Control-Aimed Kinematics, Actuation, and Sampling for a Tendon-Driven Metamorphic Hand with a Spherical Palm

Emmanouil, Evangelos

Awarding institution:
King's College London

KING'S College LONDON

University of London

Control-Aimed Kinematics, Actuation, and Sampling for a Tendon-Driven Metamorphic Hand with a Spherical Palm

by Evangelos Emmanouil

A thesis submitted in partial fulfilment of the requirements for the degree of
'Doctor of Philosophy' in Robotics

To

King's College London
University of London
School of Natural and Mathematical Sciences
Department of Informatics
Centre for Robotics Research

February 2016

I herewith declare that I have produced this PhD thesis without the prohibited assistance of third parties and without making use of aids other than those specified; notions taken over directly or indirectly from other sources have been identified as such. This PhD thesis has not previously been presented in identical or similar form to any other British or foreign examination board.

London, April 2016

To my parents, for growing me into who I am.

To my wife, who's support was paramount for the completion of this work.

And to our children, who are the future of this world.

Acknowledgments

A Ph.D. is a difficult and exciting journey where one gets to transform oneself and at the same time advance human knowledge, one step at a time. Such a journey can not be travelled alone. On a personal level, I would like to thank my friends and family for their continual support. In particular I would like to thank my parents, Ioannis and Christina Emmanouil, my father always being there and my mother always encouraging me to take that next step. My brother for all the good times. My close friends Nicholas Karavias, Miltiades Poulizos, and Spiros Papadimitriou for all the support. Finally, my wife Paraskevi and son Ioannis for everything.

I would like to thank my Ph.D. supervisor, Professor Jian S. Dai, whose advice shed a guiding light through my endeavour of obtaining a Ph.D. degree and learning what academic research is all about. He let me find my own way and when I get lost, he gently guided me back on track. A valuable ally in calm and storm. I would also like to thank my Master's degree supervisor Dimitris Gizopoulos, now an associate professor of the University of Athens and an IEEE Fellow. He was always there when I needed him the most, and is one professor who listens carefully to all the student has to say before giving an honest and valuable opinion. His passion for his subject inspired me in a remarkable way which led me to pursue a Ph.D. degree. Finally, thanks are also attributed to my B.Sc. supervisor, Professor Dimitris Dimopoulos.

Always there to help and provided excellent guidance throughout my undergraduate studies.

I also thank King's College London for the exceptional academic environment, extremely helpful staff and well designed and delivered courses of the researcher development programme. The support from the EU FP7 project TOMSY under grant No.270436 and the EU project SQUIRREL with grand No. 610532 were of immense importance and allowed me to focus my efforts on this study of metamorphic mechanisms.

Finally I would like to thank all my colleagues at King's College London, who kept me going. Always enjoyed the company and support provided by Helge A. Wurdemann and Allen Jiang. Intriguing were the discussions with Dr. Vahid Aminzadeh about control theory and Joao Bimbo on matters of robotic manipulation. Revelation always came during meetings with the two mechanism wizards, Dr Ketao Zhang and Dr. Guowu Wei. Chen Qiu and Jie Sun never failed in helping reveal paths previously unseen.

Abstract

Metamorphic mechanisms are mechanisms which have many different motion branches, regions in configuration space where the mechanism has different kinds of motion. Such mechanisms are able to transition from one branch to another to adapt to changing task requirements. This metamorphosis happens at the points in configuration space where these branches intersect, which are usually singular configurations. Analysing and controlling these mechanisms with generalised methods is usually very tedious and inefficient.

This thesis starts by examining the kinematics and singularities of one kind of metamorphic mechanisms with the aim of visualisation and control. The type of mechanisms studied are metamorphic mechanisms based on spherical linkages. In particular, the variable axis (vA) joint-based metamorphic manipulator and the KCL metamorphic hand. Then simulations, a prototype, and a carton folding operation using the metamorphic hand are presented.

First, the metamorphosis of a metamorphic manipulator is studied. The manipulator comprises three limbs, a fixed base, and a moving platform. The limbs are connected to the base and platform by the vA joint. A vA joint metamorphosis trajectory generation algorithm in the context of the metamorphic manipulator is developed. Spherical trigonometry takes centre stage in the development of this metamorphosis algorithm.

Next, the kinematics of spherical five-bar mechanisms are solved through spherical trigonom-

etry. This kind of mechanism is used as the palm of the KCL metamorphic hand. The singularities of this class of mechanisms are identified. Singularity avoidance design criteria are presented. A solution is derived for the singular but controllable configurations.

A new antagonistic tendon drive system is developed for the hand. The most important hand joints are controlled by two antagonistic tendons. The rest of the finger joints use tendons antagonised by springs and coupled with passive tendons.

Force sensitive resistors are used in the design of tendon tension sensors. These sensors are then integrated into compact tendon sensing and actuation assemblies.

Sampling time issues are identified and analysed. A CAN bus is used with the CANOpen protocol. A CANOpen protocol router is developed to reduce sampling time and control loop iteration duration.

Finally, an integrated metamorphic hand is built. Its control system is discussed and the hand is used to manipulate an articulated object, an origami-style packaging carton.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Robotics can be applied in many aspects of everyday life. Robots have traditionally been used in tasks that are repetitive, boring, and dangerous. Most robots allow for some task variation and adaptability through programming.

Expanding these capabilities is the motivation for this work. The focus is on two mechanisms capable of adapting their topology and structure to new task requirements. These two recent developments are the variable axis (vA) joint in the context of the metamorphic manipulator [1] and the metamorphic palm in the context of a metamorphic robotic hand [2, 3].

These new mechanisms are based on spherical linkages. Kinematics of spherical linkages are well studied in works such as McCarthy's [4], Ciang [5], and Duffy [6] but not in the context of metamorphic mechanisms. Further study was necessary to analyse the kinematics and singularities of spherical linkage-based metamorphic mechanisms for the purpose of

visualisation, control, and trajectory planning for reconfiguration.

1.2 Aim and Objectives

This dissertation aims to explore new methods for analysing new metamorphic mechanisms, namely the variable axis joint and the metamorphic palm. These mechanisms are used as a basis for metamorphic manipulator and hand designs. The main goal is to enable simulation of metamorphic hands and ultimately leading to the development of a reliable control system for the KCL Metamorphic hand.

The aim of this dissertation is therefore to answer the two following questions:

1 How can the kinematics of spherical linkage-based metamorphic mechanisms, namely the vA joint and metamorphic palm palm be solved, in a way useful for simulation and control, that does not fail in a case of a singular configuration.

2 How to develop a control system capable of controlling metamorphic hands that is reliable, enables position and force control, and can scale easily from a small number of joints for simpler mechanisms to a larger number of joints for dexterous hand operations.

With these aims in mind, the following four objectives are set.

1. Development of a metamorphosis strategy for spherical linkage-based metamorphic mechanisms, namely the variable axis joint and the metamorphic palm.

2. Simulation of spherical linkage-based metamorphic mechanisms, namely the variable axis joint and the metamorphic palm, and the derived metamorphic hand in the joint space and in the work space.

3. Development of a control system for a metamorphic hand which enables position or force control for each joint.

4. Demonstration of the benefits of the metamorphosis-induced dexterity and adaptability in terms of dexterous task execution with a metamorphic palm-derived robotic hand.

1.3 Thesis Structure

This thesis comprises 13 chapters, including this introduction chapter.

Chapter 2 presents a review of previous works in the field of robotic hand design and control. A number of well studied hands is presented to show how hand designs have evolved during recent times. Reviewed designs range from more focused, gripper-like hands to more complex, modular and articulated hands. A review of reconfigurable mechanisms is then followed by the introduction of the KCL metamorphic hand, and the kinematics of spherical linkages, off of which the KCL hand and the KCL variable-axis joint is based.

Chapter 3 starts by introducing the variable axis joint and the metamorphic manipulator evolved from the vA joint design. A reconfiguration strategy is developed, based on spherical trigonometry, and then a simulation is performed based on the trajectories generated by the proposed strategy. Two different control schemes are studied and compared to highlight the advantages and disadvantages of each scheme.

Chapter 4 proceeds with the further application of spherical trigonometry to the study of the kinematics of spherical linkages. Spherical linkages form the basis of the articulated palm design of the KCL metamorphic hand. Spherical trigonometry allows to easily derive the kinematic equation for spherical five bar mechanisms. It also enables differentiation of the couple-joint reflex and non-reflex configurations, similar to the elbow-up and elbow-down configurations of serial robots.

Chapter 5 goes one step further from the variable axis joint. It introduces the metamorphic palm of the KCL metamorphic hand. Singular configurations of the metamorphic palm of the

KCL metamorphic hand are identified. Based on spherical trigonometry, a set of singularity avoidance design criteria are presented. This is followed by a set of solutions for the singular but controllable cases.

Chapter 6 presents the joint space analysis of the metamorphic palm as well as the position and workspace analysis of the metamorphic hand. The joint space of the palm is generated with the singular planes and their meaning highlighted. Then the workspace of the hand is shown. Emphasis is on the benefit of having a metamorphic palm instead of a rigid one. These results are followed by a number of practical grasp examples.

Chapter 7 describes the experimental set-up used for data collection and analysis. This data was used to support the new developments relevant to the analysis and control of the KCL metamorphic hand. This enabled the later practical experiments to demonstrate the increased dexterity introduced by the metamorphic palm.

Chapter 8 discusses the practical matters related to sensing key properties of the state of the KCL metamorphic hand. The sensing data is necessary to provide feedback. This is then used by the control system to control the actuators of the hand's tendon drive system.

Chapter 9 presents the KCL metamorphic hand tendon driven control system. A number of control strategies are presented along with their advantages and limitations. The control system of the hand is broken down into multiple layers, each controlling a particular aspect of the tendon drive system. All are then combined to allow for position and force control of the hand joints.

Chapter 10 discusses the sampling time considerations for the control system of the KCL metamorphic hand. First, the parameters that effect the sampling time are analysed. Then, a solution is presented that improved the sampling time of the control system of the KCL metamorphic hand.

Chapter 11 presents the integration of all the subsystems into the final form of the KCL metamorphic hand. The relevant mechanical interfaces are presented, along with the mechanical designs used to mount the motors and other components.

Chapter 12 demonstrates the dexterity and capabilities of the hand. This is done through an experiment where the hand folds a complex origami style carton. The task is broken down into subtasks. The configurations of the hand corresponding to each subtask are shown.

Chapter 13 concludes by highlighting the main contributions of this dissertation, pointing out the limitations, and pinpointing points for future work.

Chapter 2

Review

2.1 Introduction

Robotic manipulators and robotic hands are two areas of great interest to many. Robotic applications can be found in large parts of many industries, with the most well known example being manufacturing in the automotive industry, to even household applications such as robotic vacuums and other devices.

Robotics, for the most part, are used for their versatility contrary to purpose built machinery. For assembly lines with no variation in the produced goods, it is usually best to employ purpose built machines. For assembly lines where the produced goods are very complex and change frequently, human workers are the best solution. For goods that change infrequently and involve a tedious assembly process, robotic equipment is usually employed that can be reprogrammed when the design of the produced goods change, as with most modern automotive production lines. There is a gap when the goods to be handled are complex and hazardous enough to warrant a purpose built manipulator, but also change frequently so the tooling cost

can not be justified. For these kinds of production lines, as well for tasks that require great adaptability from the robot, new kinds of mechanisms, namely metamorphic mechanisms are emerging.

This study focuses on one particular type of metamorphic mechanisms, spherical linkages, and in particular, spherical five bar linkages designed to be used as metamorphic palms for robotic hands.

2.2 Reconfigurable, Modular and Shape-Changeable Mechanisms

Metamorphic mechanisms evolved from the study of other types of reconfigurable mechanisms. The most commonly encountered types of reconfigurable but not metamorphic include general reconfigurable mechanisms, modular mechanisms and shape-changeable mechanisms.

First examples of reconfigurable mechanisms where modular designs. Hirose [7] worked on the development and control of a holonic manipulator. This is a manipulator comprising different modules with integrated processors and motors. The modules can connect and disconnect in order to change the configuration of the manipulator. Ten years later, a self-reconfiguration algorithm based on graph isomorphism and the Hungarian algorithm for the transportation problem is developed by Durna, Erkmen A.M., and Erkmen I. [8]. The authors through graph isomorphism identify the part of the manipulator that remains invariant. Then, by applying the Hungarian algorithm, they find the shortest path towards the desired configuration. Another example of a modular mechanism was designed by Chirikjian [9]. The mechanism comprised a number of identical modules. This class of mechanisms was at the time referred to as metamorphic but it is now referred to as modular. Another development in

modular mechanisms was presented by Murata, Kurokawa, and Kokaji [10]. A reconfigurable self-assembling machine was presented comprising a number of identical modules. This was later developed into the 3D self-reconfigurable structure as presented in [11].

Another type of reconfigurable mechanisms are called changeable mechanisms. These mechanisms do not change their topology or structure but are able to vary their geometry in order to better match the given task. An example of such a mechanism was designed by Wada and Asada [12]. This is a mechanism able to change its footprint. This mechanism is located at the bottom of a wheelchair and it changes the angle between the wheel axes so that it optimises the footprint of the wheeled mechanism for balancing purposes. Hong, Zhelong, Wang, and Hongwei designed a mechanism able to adapt its shape [13]. This method of reconfiguration requires a deformable external shell which is then reshaped by internal actuators.

A third type of reconfigurable mechanisms is ones that need an operator to perform the change in configuration manually. The performance of one such reconfigurable manipulator is analysed in [14].

2.3 Kinematotropic Linkages and Metamorphic Mechanisms

One of the earliest works on metamorphic mechanisms was by Wohlhart [15] on a mechanism then classified as a kinematotropic linkage. Dai started investigating metamorphic mechanisms for the packaging industry [16]. This was followed by a study of different kinds of foldable mechanisms [17]. An important work the same year was by Herve [18] who presented the lie group of rigid body displacements. Parise, Howell, and Magleby [19] presented an orthoplanar metamorphic mechanism which allowed for a topological structural change in two

orthogonal planes. Galletti and Fanghella [20] used the theory of displacement groups to derive a method for synthesizing four basic single-loop kinematotropic chains. The same method is then extended to multi-loop kinematotropic chains. Liu studied and explained the essence and characteristics of metamorphic mechanisms and their metamorphic ways [21]. Dai and Jones [22] then presented a new matrix representation of the topological change of metamorphic mechanisms.

One important aspect of mechanism analysis, directly related to control, is mechanism mobility. It is essential to know the mobility of a mechanism which then drives the decision on how the mechanism needs to be actuated. The well-known Grubler-Kurtzbach criterion has its limitations, especially in parallel mechanisms. Rico, Gallardo and Ravani [23] presented a method for mobility analysis of various kinematic chains based on the lie algebras of the Euclidean space. This work was later extended by Rico el at. [24] to a mobility criterion for parallel manipulators more general than the Grubler-Kurtzbach criterion. This was based on the sub algebras of the Lie Algebra, $se\,(3)$. The problem of mobility analysis was also studied by Dai, Huand and Lipkin in the same year [25]. The authors presented a new method for performing mobility analysis of over-constrained parallel manipulators based on decompositions of motion and constraint screw systems.

Metamorphic mechanism representation was further developed by Yan and Kuo [26]. A class of metamorphic mechanisms named "variable kinematic joints" and their topological representation was presented. A number of mechanisms ranging from simple reconfigurable joints to complex lock mechanisms and toys are analysed by means of graph theory and methods similar to those of finite state machines. In [27], the mobility and configuration singularity of mechanisms with variable topologies was investigated by utilising screw coordinates.

In 2008 and 2009, a number of interesting developments occurred. Zhang, Wang and

Dai [28] presented a metamorphic mechanism synthesis method based on evolution principles. Then in 2009, Zhang and Dai used the Lie displacement subgroup for the purpose of reconfiguring spatial metamorphic linkages [29]. In the same year, foundational work by Liping and Dai set the basis for reconfiguration and metamorphosis [30]. They also presented a number of different kinds of metamorphic mechanisms and their associated reconfiguration principles. At the same time they presented the fundamental definitions of configuration, topology and topological structure in the context of metamorphic mechanisms [31]. Further work on the representation and reconfiguration of metamorphic mechanisms was performed by Ting-li et al in [32]. A way to represent a mechanism by using a Position and Orientation Characteristic matrix and then three ways of reconfiguring the mechanism were presented. Latest developments in metamorphic mechanism representation include the work done by Zhang [33]. A new representation of reconfigurable linkages is presented utilising binary strings and thus enhancing the traditional adjacency matrix into a three-dimensional array.

2.4 Metamorphic Mechanism Examples and Applications

A number of metamorphic mechanisms have been developed in the last decade. One example is a class termed "variable kinematic joints" [26]. The concept of variable kinematic joints was further developed by K. Zhang and led to the development of the parallel metamorphic manipulator appearing in [34]. Leal and Dai presented another class of 3DOF mechanisms with a 3-RPRP architecture [35].

One example of applying the concept of metamorphic mechanisms on an already known type of mechanisms is the King's College London (KCL) metamorphic hand. In particular, the metamorphic hand comprises a metamorphic palm and simple tendon driven fingers. It was first developed and analysed by Wang and Dai [36, 37]. The workspace of the three fingered

metamorphic hand was later analysed and presented [38]. The three fingered metamorphic hand evolved into a five fingered hand and its kinematic analysis was presented by Wei, Dai, Wang, and Luo [3].

Another example of a metamorphic mechanism based on the principle of annexing links was a walking robotic chair by Zhengyan, Hongbo, Zhen, and Lili [39]. This mechanism can change its topology from a quadruped to a biped robot by connecting the lower parts of each leg.

Kinematotropic linkages were the basis for metamorphic mechanisms so it was natural for some metamorphic linkages to appear. A novel 8-bar mechanism was developed by Guowu and Dai based on an origami carton [40]. This metamorphic 8-bar linkage led to K. Zhang and Dai developing an over constrained 6R linkage [41].

An extension to Dai's work for the packaging industry [16] is presented in [42]. The authors investigated the stiffness of folded carton packaging. Another important development was the introduction of metamorphic process for manufacturing by Carrol et al. in 2005 [43].

2.5 Reconfigurable Robots and the New Metamorphic Manipulator

Robots with reconfigurability are a class of robotic systems that can change their topology and functionality to adapt to unexpected environments and conduct complicated tasks. This kind of robots have improved flexibility and adaptability compared to those with invariant topological structure. This raises challenges in both the fundamental basis and the advancement of this technology [16, 44, 45, 46]. Much interest has arisen to address these challenges in the field of mechanisms and robotics and new reconfigurable robots are widely used in do-

mestic, hazardous, aerospace, manufacturing and medical applications [47, 48]. Metamorphic mechanisms [49] and kinemetatropic linkages [15, 50] are representative developments which match the ever-growing demand of reconfigurable devices for various applications.

Metamorphic mechanisms are a class of mechanisms that have the ability to change their topological structure from one configuration to another. The resultant change of motion characteristics are in the form of mobility variation and a change of motion mode, leading to mobility change. Starting from the investigation of metamorphosis embodied in origami folding and packaging cartons [16], the reconfigurability of this kind of mechanisms has been broadly studied. Elementary matrix operations were introduced to describe the topological configuration transformation in metamorphic mechanisms [22]. In order to preserve the original information of the metamorphic mechanism in the process of topological change, an approach for topological representation of mechanisms was developed based on a three-dimensional matrix with integrated 16 bit words [51].

Compared to new developments in reconfigurable mechanisms and robots, most parallel mechanisms have invariant topology and a monotone motion mode. This leads to restricted workspace and dexterity of the end-effector [52, 53, 54]. A multi-operational parallel mechanism with different operation modes separated by constraint singularity was identified in the study of constraint singularities [55] and 3-DOF parallel mechanisms with both spherical and translational motion modes were synthesized [56]. Inspired by the inherent principle of reconfigurability, a variable-axis (vA) [57] joint extracted from origami folds was designed. The topological reconfiguration and the bifurcated motion characteristics of the evolved parallel mechanisms were analysed [58]. Other relevant works are the two types of metamorphic parallel mechanisms which employ the new rT joint [59, 60] by integrating an extra rotational degree of freedom in the well-known Hooke's joint.

2.6 Robotic Hand Designs and the KCL Metamorphic Hand

Dexterous robotic hands are a topic that attracted a lot of attention since the milestone development of the Stanford/JPL hand and the MIT/Utah hand in the early of 1980s [61]. Since then, a number of multifingered robotic hands have been designed and developed all over the world. The prosthetic hand-based Belgrade/USC hand [62] incorporated a thumb and two more coupled pairs of fingers that adapted to the shape of the grasped object and so the whole hand needed only four motors while it had five digits. The highly integrated DLR-Hand [63] incorporates purpose built linear actuators, position sensors both for the motors and joints, tactile sensors on each finger link, a stereo camera on the palm, and two axis torque sensors at the finger tips. The Robonaut hand [64, 65] is designed to be similar in size and capability to an astronaut's hand in a suit and withstand the environment of space. The tendon driven Shadow Robot Hand [66] has a one DOF articulated palm and a structure closely resembling the human hand with the option to either use electric motors or pneumatic artificial muscles. The self-contained three-fingered Barrett hand [67] with one finger fixed on the palm and two fingers able to rotate around the palm. The low-cost easy-to-use LARM hand [68, 69] with three one DOF fingers having each finger's joints coupled by four-bar linkages designed to mimic a human performing a cylindrical grasp. The UB hand [70, 71] has explored many novel control and actuation concepts including a twisted string actuator where by twisting two strings a rotary motion is converted to a linear one. The DLR/HIT II hand [72] consists of a palm module and finger modules with actuators and control system integrated in each module. The SKKU hand [73] also uses identical finger modules each with it's own integrated control system and motors as well as six DOF force-torque sensors at the fingertips. All these hands,

anthropomorphic or non-anthropomorphic, are capable of performing dexterous motions to a certain degree. However, they are either based on a rigid palm, or the degree to which the palm can be articulated is limited to splitting the palm into two or three sections. As such, the workspace and dexterity of these hands are limited.

A novel metamorphic robotic hand was invented and developed by Dai and his colleagues [3, 37, 74, 75] based on the concept of metamorphosis [17] stemming from origami folding [22, 76]. This design aimed to achieve a greater workspace and additional dexterity and versatility. The novelty of the metamorphic robotic hand lies in the introduction of an articulated palm. The palm is formed of a spherical metamorphic linkage that grants the palm the ability to change its mobility, topology, and configuration. This articulated palm design enables the new robotic hand to perform and emulate more complex and sophisticated hand motions. It also allows the hand to fold and pass through tight spaces and to change its configuration to adapt to various task requirements. Therefore, a thorough investigation of the articulated palm plays a critical role in understanding the kinematics, dynamics, control, and applications of the novel metamorphic hand.

2.7 Kinematics of Spherical Linkage-Based Metamorphic Mechanisms

According to many textbooks [6, 77, 78, 79, 80], the most popular methods for kinematics modelling are the loop closure method, vector loops method, Denavit-Hartenberg method and the method of successive screw displacements.

Dynamics modelling is usually done either by the recursive Newton-Euler method, by applying the Euler-Lagrange equations of motion or by using screw algebra [79, 80]. In both

the velocities of the centre of mass and the inertia tensors for each rigid body are needed.

Spherical linkages, linkages with the property that every link in the system rotates about the same fixed point, have been widely investigated and used by mathematicians, mechanicians, and engineers. Liu and Ting [81] established rotatability criteria for spherical chains based on rotatability laws for planar linkages. Gosselin and Hamel [82] developed the agile eye based on a spherical parallel mechanism. In order to investigate the kinematics of spherical mechanisms, Wampler [83] formulated loop equations to solve the kinematics of parallel spherical mechanisms up to three loops. McCarthy [4] used structure equations together with trigonometric constraints to analyse the kinematics of serial and closed-loop spherical linkages. Gupta and Beloiu [84] proposed algebraic-geometrical methods to eliminate branch and circuit defects in the synthesis of spherical four-bar linkages. Chiang [5] has also carried out significant work on the analysis of spherical mechanisms but primarily focuses on spherical four-bar linkages. Duffy [6] used spherical trigonometry to solve geared five-bar linkages of mobility $M = 1$ and provided the inspiration that lead to the work presented in this dissertation.

Using loop equations Cui and Dai [2], and Wei et al. [3] investigated kinematics, workspace and manipulability of the metamorphic robotic hand. However, the work presented in [2, 3] focused only on the direct or forward kinematics of the metamorphic robotic hand. Further, there was no differentiation of the reflex and non-reflex joint-coupler configurations. By using spherical trigonometry, a solution for the forward and inverse kinematics of the metamorphic palm is derived. This solution does not involve solving a system of complicated trigonometric equations that are difficult to understand intuitively. Thus, the process is less prone to human error compared to solving a system of a large number of equations, such as the one produced when using quaternions, the D-H method or the loop closure method. Further, the proposed method does not involve quadratic equations which leads to loss of the sign information of the

joint angles, thus making identification of the reflex and non-reflex joint-coupler configurations difficult. By using spherical trigonometry, this dissertation presents an intuitive method that preserves joint angle sign information for workspace and joint space analysis of the reconfigurable palm of the KCL metamorphic hand. In turn, this provides insights into work on path planning and control for the metamorphic hand.

2.8 Conclusions

To develop new metamorphic mechanisms and robotic hands, it is important to study spherical linkages in the context of metamorphic mechanisms. Existing studies on spherical five-bar linkages and parallel manipulators do not take into consideration the need to differentiate the coupler reflex and non-reflex configurations. Studies in trajectory planning for reconfiguration in the context of the new vA joint are limited, because the vA joint is such a new development.

The fore-mentioned metamorphic mechanisms have a common point. They are all based on spherical mechanisms. By using spherical trigonometry their properties can be studied with the simplicity intrinsic to spherical trigonometry-constrained geometry contrary to generalised methodologies. Spherical trigonometry-based kinematics, singularity analysis and trajectory planning for reconfiguration are simple to follow. Through this study valuable insights are gained on the properties of spherical mechanism-based metamorphic manipulators.

Chapter 3

Spherical Trigonometry-Based Trajectory Generation for Metamorphosis

3.1 Introduction

A spherical mechanism-based variable axis (vA) joint was designed by Dr. Jiang to be used in a metamorphic parallel manipulator. This manipulator can change its mobility from three to four to five to six degrees of freedom (DOF). The vA joint has a source configuration where it behaves as a spherical joint. From the metamorphosis source configuration, it can be transformed into a two DOF joint, referred to as a universal joint, u-joint or Hooke joint. The vA joint can also be transformed into a one DOF, revolute joint equivalent.

For the vA joint to transform between the three configurations, while being part of a

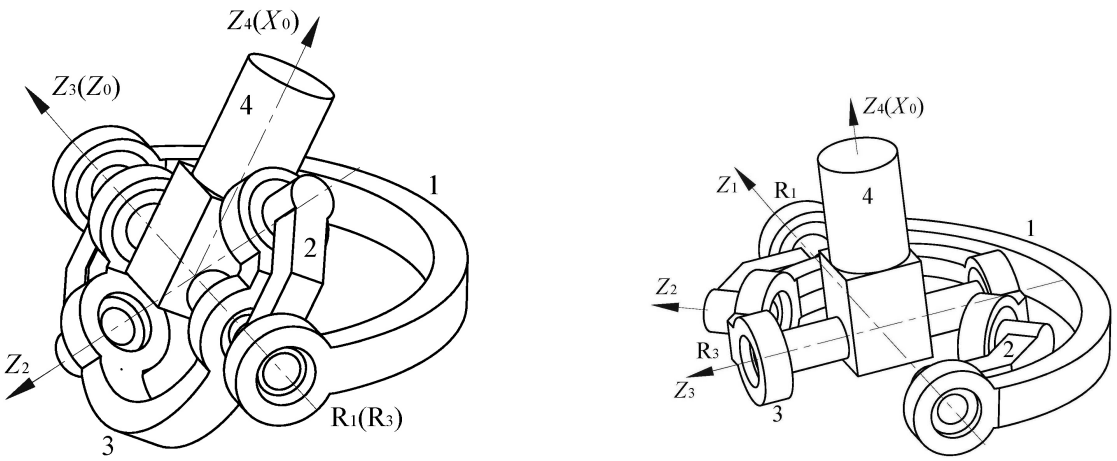
metamorphic parallel manipulator, it was found excessive forces were necessary if a linear trajectory profile was followed. Further, it was not always possible to start the metamorphosis, depending on the current configuration of the joint. It was thus necessary to further study the joint and identify an appropriate trajectory tracking algorithm for reconfiguration.

For the purpose of on-line reconfiguration of the metamorphic parallel manipulator, a reconfiguration strategy is proposed. From this proposed strategy, a task constraint emerges and the mechanism is then treated as a redundant parallel manipulator. First, the task is planned with an independent variable. Then position and velocity profiles are generated for each actuator, taking into account the initial and final topological configuration as well as motor maximum velocities. A similar method has been employed in [85] where first the values of some independent variables are determined and then the parameters of the actuator displacements are derived based on the independent parameters. Spherical trigonometry is key in the metamorphosis trajectory generation.

3.2 The Variable-Axis Joint

The variable axis joint is a spherical linkage-based metamorphic mechanism that can change from a spherical joint equivalent (R_s) to a universal joint equivalent (U_e) or pin or revolute joint equivalent (R_e).

The vA joint reduces its link number by annexing link 3 to link 2 as the configuration in Fig. 3.1b and the motion of the output link 4 degenerates to two independent rotational motions equivalent to Hooke's pair. The vA joint changes into the second stable subphase in Fig. 3.1a when link 3 and link 2 are annexed in the typical configuration where the joint axes R_1 and R_3 become coincident. This results to the equivalent compound rotating pair and the output link 4 has one rotational motion with respect to the base link 1.



(a) Metamorphosis in the Initial Appearance (R_e). (b) Metamorphosis in the Final Appearance (U_e).

Figure 3.1: Variable Axis Joint Metamorphosis.

3.2.1 The Variable-Axis Joint Evolved Metamorphic Parallel Manipulator

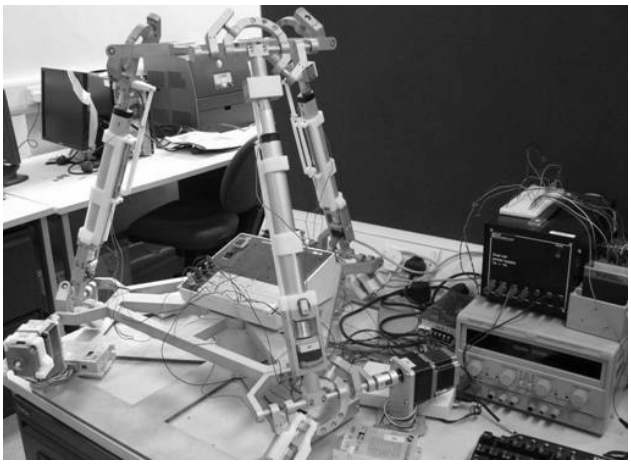


Figure 3.2: Prototype of the Metamorphic Parallel Manipulator.

The metamorphic parallel manipulator, as shown in Fig. 3.2, is a symmetrically constructed manipulator. It comprises a movable platform and three limbs. The limbs are constructed

by integrating the vA joint at both ends of the prismatic joint in each limb [1]. The axes of revolute joints connecting to the base are coplanar. The common points of the three S_e pairs are connected to the base plane and form the equilateral triangle which is the base. Similarly, the axes of revolute joints connecting to the platform are coplanar and the common points of the three S_e pairs connected to the platform form the equilateral triangle which is the platform.

3.3 Control Strategy for Reconfiguration

The objective of the proposed strategy is to prevent platform motion during reconfiguration. By preventing platform motion the planing task is kept simple. Further no forces are generated, associated with a moving platform. This is achieved by not moving the platform during metamorphosis. The initial (R_e) and final (U_e) configurations are seen in Fig. 3.1a and Fig. 3.1b. The starting and ending angular displacements for the first input joint, θ_1 are 0 and $\pi/2$. The starting and ending displacement for the second input joint, θ_2, are π and 0.

Without the task constraint, the endpoint of the output link moves on a S^2 manifold of R^3 and the mechanism is a mobility 3 mechanism. The task constraint reduces the degrees of freedom to 1 by introducing a position constraint. This implies that task planning can happen in a one dimensional space. The unconstrained task variable is the orientation of the output link in S^2, angle θ_4.

3.4 Trajectory Planning for a Single Variable Axis Joint

One method to analyse the vA joint, is to assign the frame as seen in Fig. 3.1, then compute the appropriate Denavit-Hartenberg parameters and finally, apply the task constraint seen in

Eqn. (3.1).

$$c_x = 0 \tag{3.1}$$

Where $c_x = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \boldsymbol{c}$, the X coordinate of vector $\boldsymbol{c}$. Vector $\boldsymbol{c}$ is the vector corresponding to the revolute joint at the intersection of the Z_3 axis and link 3. However, since the vA joint is a spherical mechanism the D-H parameters are not necessary and rotation matrices can be used instead of homogeneous transform matrices. Thus, Eqn. (3.2) can be written for $\boldsymbol{c}$.

$$\boldsymbol{c} = R\left(z_1, \theta_1\right) R\left(y_1, \alpha_2\right) R\left(z_2, \theta_2\right) R\left(y_2, \alpha_3\right) \begin{bmatrix} 0 \\ 0 \\ r_3 \end{bmatrix} \tag{3.2}$$

Where $R\left(z_i, \theta_i\right)$ and $R\left(y_i, \alpha_i\right)$ are rotation matrices of angle θ_i about axis z_i and angle α_i about axis y_i. Finally, r_3 is the distance of joint 3 from the center of the spherical mechanism. For simplicity we choose $r_3 = 1$. By performing matrix multiplications, Eqn. (3.2) yields Eqn. (3.3).

$$\begin{aligned} c_x = \cos\theta_1 \left(\cos\alpha_1 \sin\alpha_2 \cos\theta_2 + \sin\alpha_1 \cos\alpha_2\right) \\ - \sin\theta_1 \sin\alpha_2 \sin\theta_2 \end{aligned} \tag{3.3}$$

Solving Eqn. (3.3) for θ_1, Eqn. (3.4) is derived.

$$\theta_1 = \sin^{-1}\left(\frac{1 + \cos\theta_2}{\sqrt{3 - \cos^2\theta_2 + 2\cos\theta_2}}\right) \tag{3.4}$$

Equation (3.4) can be used in a cross-coupled control scheme to make sure that the reconfiguration constraint is maintained throughout task execution. However, one important

observation can lead to a simplified solution. All joint axes intersect at the center of the vA joint. Thus, the metamorphic joint is a spherical mechanism. Moreover, since the angular displacements of the task are limited to $max(\theta_i) = \pi$, it makes sense to apply spherical trigonometry.

As seen in Fig. 3.3, a unit sphere concentric to the mechanism is drawn. The frame of the sphere has it's X axis collinear with the Z_0 axis and it's Y axis collinear with the X_0 axis. Each joint axis intersects with the surface of the sphere in two points. For simplicity, only one hemisphere is concerned. Point A is the intersection of the Z_1 axis and the sphere, point B is for Z_2 and point C is for Z_3. The output link originates at the center of the S^2 sphere and is normal to $\triangle OAC$. The joint axes Z_i are chosen originating from the center of the sphere and pointing outwards, towards the points A, B and C. Finally, Z_4 is pointing downward. This way they are compatible with spherical trigonometry laws.

The cosine law for spherical triangles on the triangles seen in Fig. 3.3, states:

$$\cos AC = \cos AB \cos BC + \sin AB \sin BC \cos ABC \tag{3.5}$$

$$\cos BC = \cos AB \cos AC + \sin AB \sin AC \cos BAC \tag{3.6}$$

According to the mechanism geometry, Eqn. (3.8), (3.9), (3.10) apply.

$$AC = \theta_4 \tag{3.7}$$

$$\theta_1 + BAC = \pi/2 \tag{3.8}$$

$$\theta_2 + ABC = \pi \tag{3.9}$$

$$AB = BC = \pi/4 \tag{3.10}$$

After combining Eqn. (3.5), (3.6) with Eqn. (3.7), (3.8), (3.9), (3.10) and solving for θ_1

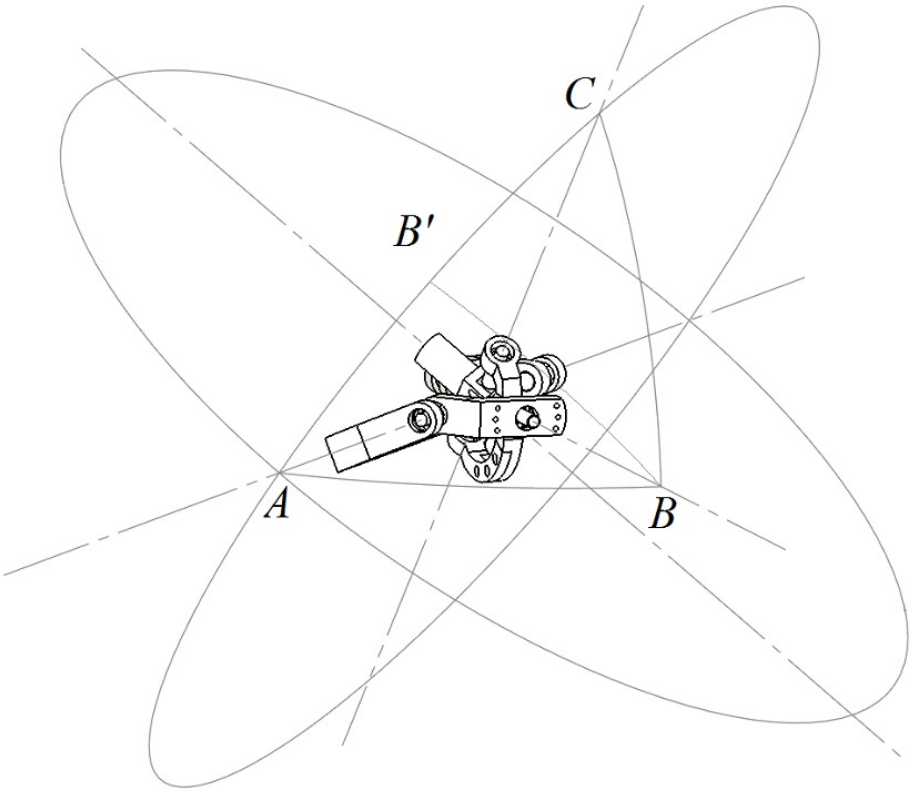


Figure 3.3: The Metamorphosis of the vA Joint.

and θ_2, Eqn. (3.11) and Eqn. (3.12) are derived.

$$\theta_1 = \sin^{-1}\left(\frac{1-\cos\theta_4}{\sin\theta_4}\right) \tag{3.11}$$

$$\theta_2 = \cos^{-1}(1-2\cos\theta_4) \tag{3.12}$$

By differentiating Eqn. (3.11) and Eqn. (3.12), the relations in Eqn. (3.13) and Eqn. (3.14) can be observed between the mechanism joint space (θ_1, θ_2) and θ_4.

$$\dot{\theta_1} = \frac{1-\cos\theta_4}{\sin\theta_4\sqrt{2\left(\cos^2\theta_4-\cos\theta_4\right)}}\dot{\theta_4} \tag{3.13}$$

$$\dot{\theta_2} = \frac{\sin\theta_4}{\sqrt{\cos^2\theta_4-\cos\theta_4}}\dot{\theta_4} \tag{3.14}$$

$$\tag{3.15}$$

Equations (3.13) and (3.14) can be combined into a Jacobian matrix mapping from θ_4 to the mechanism joint space (θ_1, θ_2).

$$\dot{q} = J\left(\theta_4\right)\dot{\theta_4} \tag{3.16}$$

Where:

$$\dot{q} = \begin{bmatrix} \dot{\theta_1} \\ \dot{\theta_2} \end{bmatrix} \tag{3.17}$$

$$J\left(\theta_4\right) = \begin{bmatrix} \frac{1-\cos\theta_4}{\sin\theta_4\sqrt{2(\cos^2\theta_4-\cos\theta_4)}} \\ \frac{\sin\theta_4}{\sqrt{\cos^2\theta_4-\cos\theta_4}} \end{bmatrix} \tag{3.18}$$

Finding a trajectory in task space, is now straightforward. The trajectory of the output link's rotation about the Z_4 axis in task space is a straight line from 0 to $\pi/2$. Since the control system is a digital system and each motor has a maximum attainable speed, task space trajectory generation for a single metamorphic joint can be done as follows.

By knowing the maximum velocity of each motor, the maximum velocity for θ_4 can be computed for a given position according to Eqn (3.19) and Eqn (3.20), which are derived from Eqn. (3.13) and Eqn. (3.14).

$$\dot{\theta_{4a}} = \frac{\sin\theta_4\sqrt{2\left({\cos\theta_4}^2-\cos\theta_4\right)}}{1-\cos\theta_4}max(\dot{\theta_1}) \tag{3.19}$$

$$\dot{\theta_{4b}} = \frac{\sqrt{{\cos\theta_4}^2-\cos\theta_4}}{\sin\theta_4}max(\dot{\theta_2}) \tag{3.20}$$

Since we know the time constant of the system (it can be determined by the frequency of the control loop), the next position for θ_4 can be computed by multiplying the previously computed velocity by the time constant and adding to the current position. This method will generate a trajectory comprising positions and velocities for θ_4. These correspond to the system moving as fast as possible while maintaining task restrictions. Pseudo code for the above algorithm can be seen in Fig. 3.4.

finished = false;

while not finished

$\dot{\theta_{4a}} = \frac{\sin\theta_4\sqrt{2(\cos{\theta_4}^2-\cos\theta_4)}}{1-\cos\theta_4} max(\dot{\theta_1})$;

$\dot{\theta_{4b}} = \frac{\sqrt{\cos{\theta_4}^2-\cos\theta_4}}{\sin\theta_4} max(\dot{\theta_2})$;

new$\dot{\theta_4}$=min($\dot{\theta_{4a}}$,$\dot{\theta_{4b}}$)

new θ_4 = timeConstant * new$\dot{\theta_4}$+lastθ_4;

θ_4 = [θ_4 newθ_4];

$\dot{\theta_4}$ = [$\dot{\theta_4}$ new$\dot{\theta_4}$];

lastθ_4 = newθ_4;

if last $\theta_4 > \pi/2$-threshold

finished = true;

end

end

Figure 3.4: Algorithm for Trajectory Planning.

One important thing to notice is that the starting as well as the final positions are singular configurations. Because of that, when the metamorphic joint reaches those configurations, the

desired speed is set to $\dot{\theta}_i = 0$ for the final configuration and equal to the next non-singular value for the initial configuration.

After generating velocity and position profiles for θ_4, the planned trajectory can be mapped to the vA joint space by using the Jacobean matrix from Eqn. (3.18) as well as Eqn. (3.11) and (3.12).

3.5 Implementation Of The Trajectory Planning Algorithm for a Single VA Joint

A program was developed to perform trajectory planning as proposed in this chapter. Both motors are assumed to have a maximum absolute velocity of $\pi/2 rad/sec$ and a control loop frequency of $100Hz$, thus a time constant of $0.01sec$. The planned trajectories in relation to θ_4 are seen in Fig. 3.5. The planned position profile is in Fig. 3.6 and the velocity profile is in Fig. 3.7.

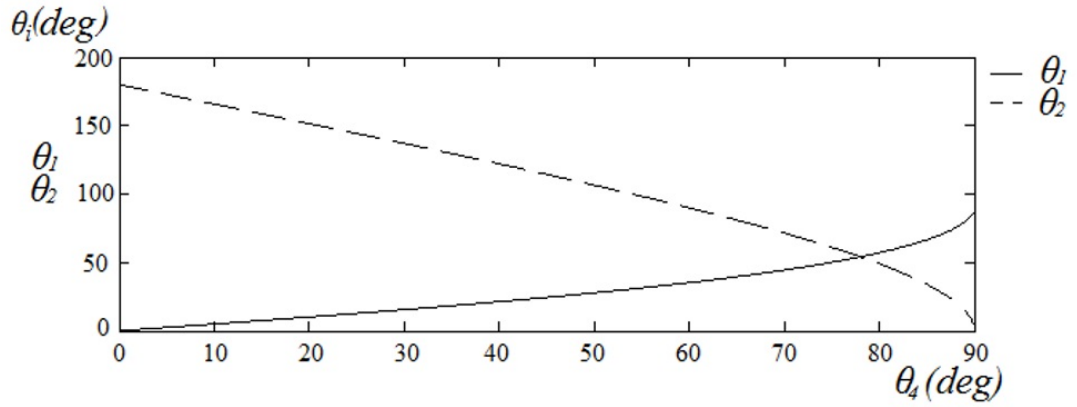


Figure 3.5: Planned Joint Angle Relationship.

As discussed, one of the motors should be planned to rotate at its maximum velocity. As seen in Fig. 3.7, motor 2 is that motor since it's velocity is constant throughout task execution.

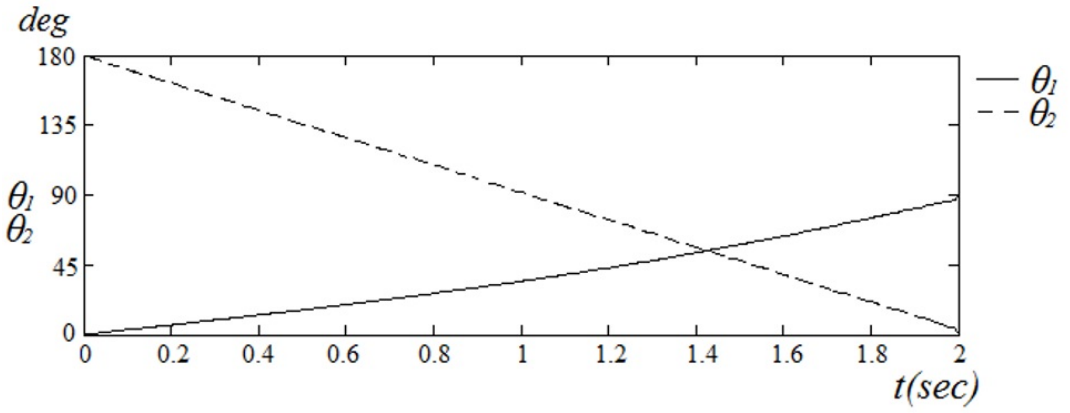


Figure 3.6: Position Profile.

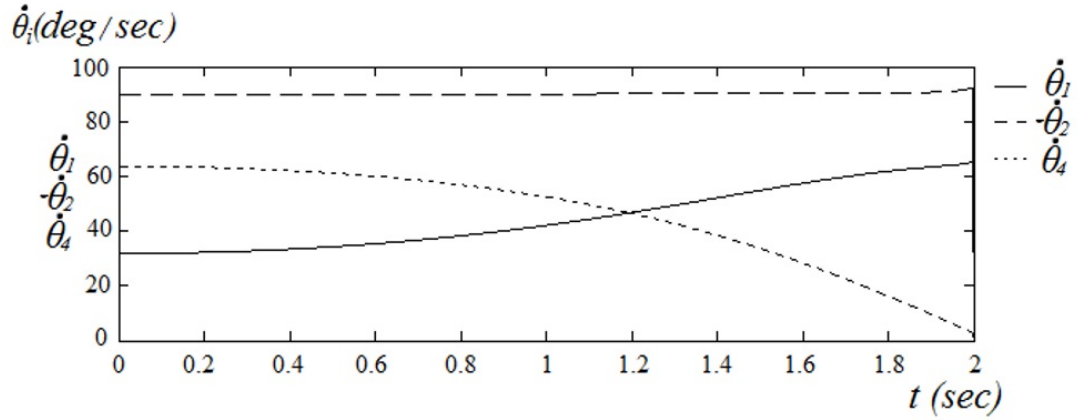


Figure 3.7: Velocity Profile.

As a consequence of the slope of θ_2 being constant, θ_4 has to change in a non-linear fashion. Accelerations on θ_1 are also smaller than if θ_4 changed linearly.

3.6 Simulation Results

Simulations are performed by attaching simulated motors to the vA joint model and set to follow the reference position and velocity profiles. Joint angular position, velocity and accel-

eration for the input and output joints as well as the coordinates of point C are recorded. The collected data is then analysed by computing average absolute deviation (AAD) and maximum absolute deviation from the reference trajectories.

Figure 3.8 is the graph of planned and simulated trajectories for θ_1 and θ_2 in the position control case. For the position control case, position tracking for θ_1 has a maximum error of 2.54 deg and an AAD of 0.06 deg. For θ_2 the maximum absolute deviation is 3.83 deg and the AAD is 0.16 deg.

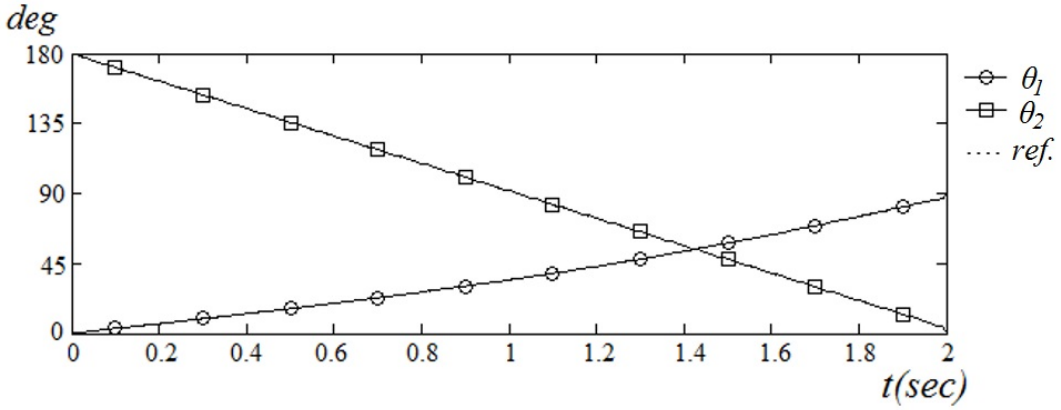


Figure 3.8: Tracking Joint Angles with Position Control.

Figure 3.9 is for the velocity control case. Position tracking for θ_1 is achieved with a maximum error of 1.96 deg and an AAD of 0.81 deg. For θ_2 the maximum absolute deviation is 4.08 deg and the AAD is 1.79 deg. In Fig. 3.10, the graph of the position of point C of the vA joint in $X_0Y_0Z_0$ can be seen.

For clarity, the X coordinate of point C in $X_0Y_0Z_0$ frame can be seen in Fig. 3.11 with a different scale than that of Fig. 3.10. The task is defined as $c_x = 0$. The dotted line depicts the desired trajectory, the dashed line is the simulated trajectory with velocity control and the solid line is for the position control case.

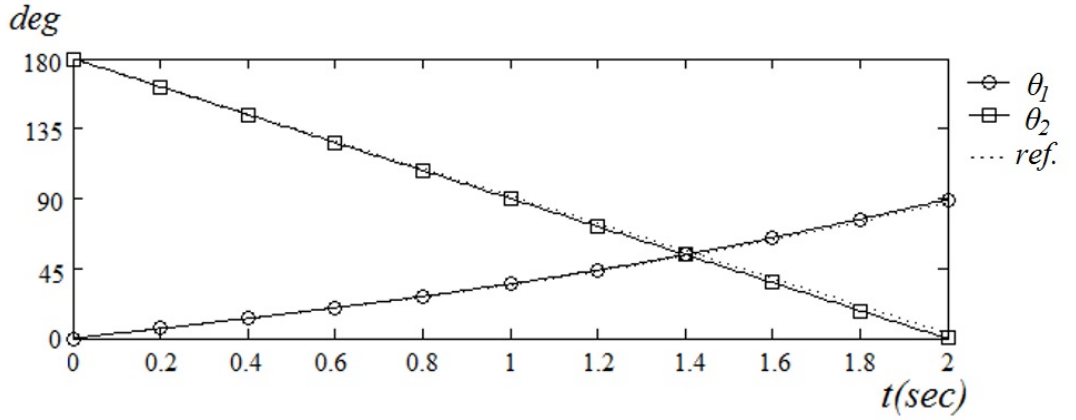


Figure 3.9: Tracking Joint Angles with Velocity Control.

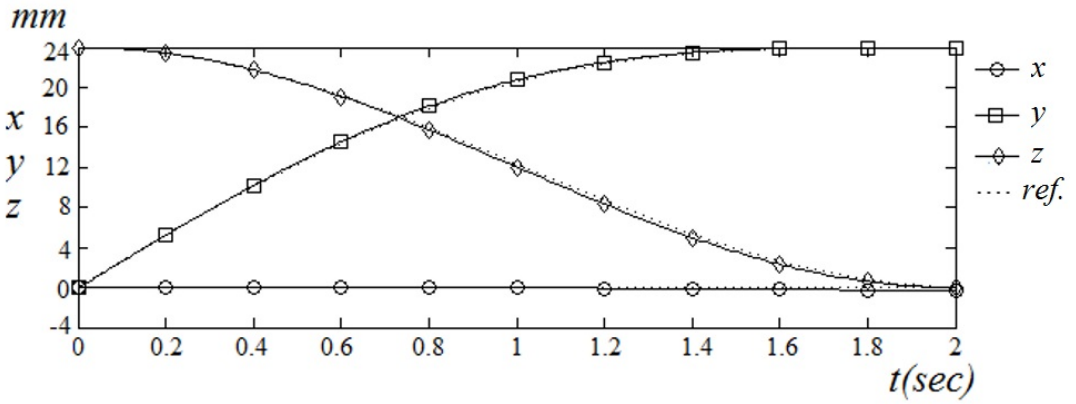


Figure 3.10: Tracking of Point C In $X_0Y_0Z_0$.

While using velocity control, the task is achieved with a maximum absolute deviation of $0.28mm$ and an AAD of $0.08mm$. Position control exhibited an improved maximum deviation of $2.8\mu m$ and an AAD of $0.7\mu m$. The reason for the poor performance of pure velocity control is the digital nature of the system in combination with the inability of the current hardware to have acceleration control. Since velocity remains constant between two sampling periods, the tracking error increases as the control frequency becomes smaller or the motors move faster.

Although position control was more accurate, velocity control exhibited smoother trajec-

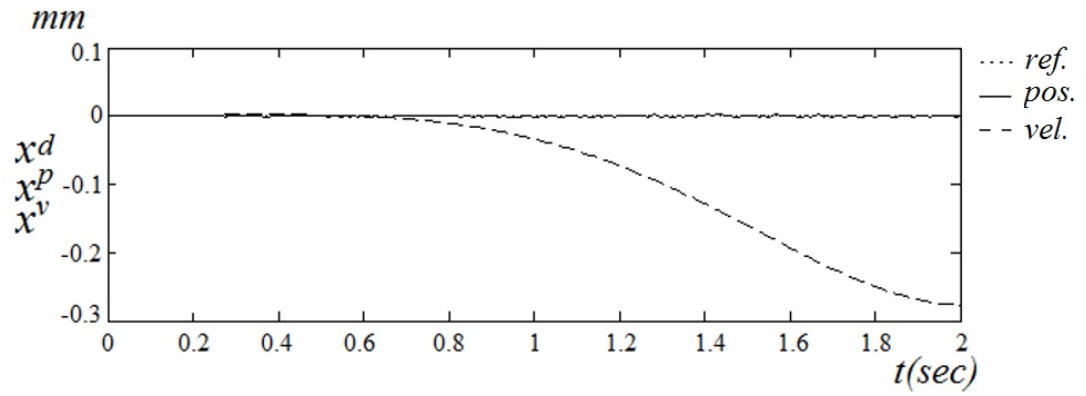


Figure 3.11: X Coordiante of Point C in $X_0Y_0Z_0$.

tory tracking. Figure 3.12 shows the X coordinate of point C in $X_0Y_0Z_0$ frame in a scale chosen to make the deviation from 0 obvious. Simulation data is represented by a solid line and a dotted line is the desired trajectory.

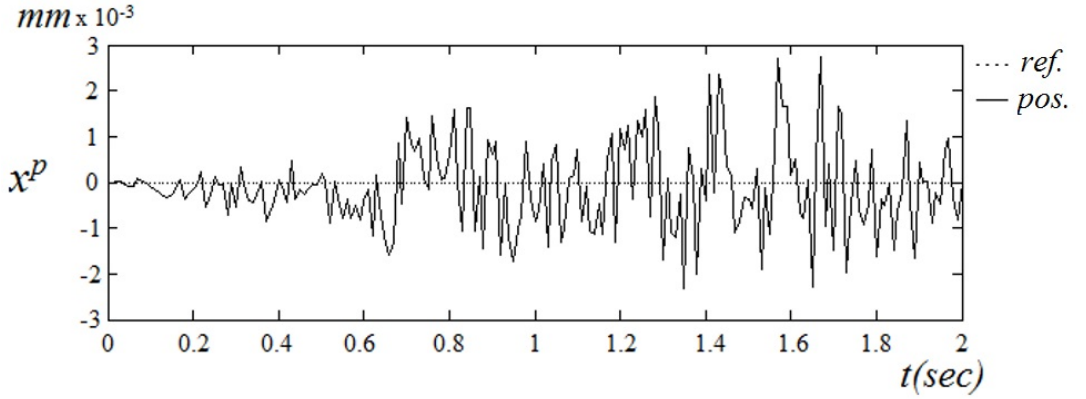


Figure 3.12: c_x Deviation with Position Control.

A ninth order polynomial best fitted the data in the velocity control case. The velocity control trajectory was smoother with an AAD of $0.01\mu m$ from the best fit, two orders of magnitude smaller than the position control case.

Finally, a simulation was performed to compare the benefit of planning and tracking tra-

jectories instead of just having a linear motion. The focus is on the displacement of point C along the X axis of the $X_0Y_0Z_0$ frame. The results can be seen in Fig. 3.13. The dotted line is the desired trajectory for c_x. The dashed line represents c_x in the velocity control case. The solid line is the trajectory of c_x. The motors rotate with a constant velocity, defined by the initial and final positions for each joint. The maximum error in the constant velocity case is $3.84mm$ with an average absolute deviation of $2.11mm$.

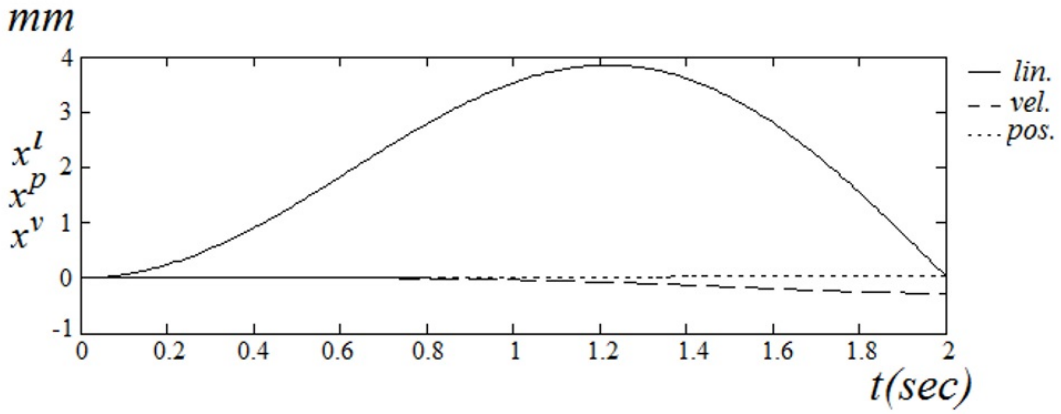


Figure 3.13: Trajectory Planning vs. Linear Motion.

The maximum error in the case of linear motion is two orders of magnitude greater than if using the proposed trajectories. This shows that planning and following trajectories based on the task constraint is better than just using a control scheme with a constant speed setting and set points for the final configuration. These results are summarized in Tab. 3.1.

3.7 Conclusions

In this chapter a new reconfiguration strategy and trajectory planning algorithm for the metamorphosis of the variable axis joint, when used in the metamorphic parallel manipulator, was developed. The central principle of the algorithm is to actuate the joint in such a way so the

Table 3.1: Tracking Average and Maximum Error.

Variable	Deviation			
	Max (mm)	Max (deg)	AAD (mm)	AAD (deg)
θ_1 position ctrl.	-	2.54	-	0.06
θ_1 velocity ctrl.	-	1.96	-	0.81
θ_2 position ctrl.	-	3.83	-	0.16
θ_2 velocity ctrl.	-	4.08	-	1.79
c_x constant vel.	3.84	-	2.11	-
c_x position ctrl.	$2.8x10^{-3}$	-	$0.7x10^{-3}$	-
c_x velocity ctrl.	0.28	-	0.08	-
c_x fit vel. ctrl.	$0.4x10^{-3}$	-	$0.01x10^{-3}$	-

output link of the joint only rotates about it's principal axis. This allows for reconfiguring the metamorphic parallel manipulator from the 3-RPS configuration to the 3-UPS configuration and back. There is no motion in the moving platform and thus controller effort is drastically reduced.

Simulations verified the suitability of the task constraint for reconfiguring the vA joint. Trajectories were generated and a simulation was performed which demonstrated correct task execution. The results of the simulation showed that pure velocity control exhibited a small tracking error due to the digital nature of the control hardware. Moreover, position control yielded better results but the motion was not as smooth as the velocity control method.

Chapter 4

Spherical Trigonometry-Based Kinematics

4.1 Introduction

The novelty of the metamorphic hand lies in the introduction of the articulated/reconfigurable palm. This palm enables the multi-fingered robotic hand to have a greater range of grasping poses. Therefore, a thorough investigation of the palm plays a critical role in the further exploration of the kinematics, dynamics, control and applications of the hand. There have been a number of works on kinematics of spherical mechanisms. However, the existing solutions either do no differentiate the joint-coupler reflex and non-reflex configurations or are very complex and do not provide an intuitive insight into the nature of the mechanism by inspection of the equations. This chapter focuses on the kinematics issue of the metamorphic palm, based on spherical trigonometry. The method presented herein makes a distinction of the coupler-joint reflex and non-reflex configurations. Its simplicity also simplifies singularity analysis, as

can be seen in the following chapter 5.

First, the positions of all but the joint-coupler are calculated. Then, all chords and spherical links of the palm are computed. The palm is segregated into spherical triangles. These spherical triangles are solved and all joint angles, including the joint-coupler angle, are calculated. This is done both for the joint-coupler reflex and non-reflex configurations. Finally, the position of the joint-coupler and the coupler links are computed.

4.2 Geometry and Coordinate Systems

First, it is important to define the relevant coordinate systems and how they relate to each link of the mechanism.

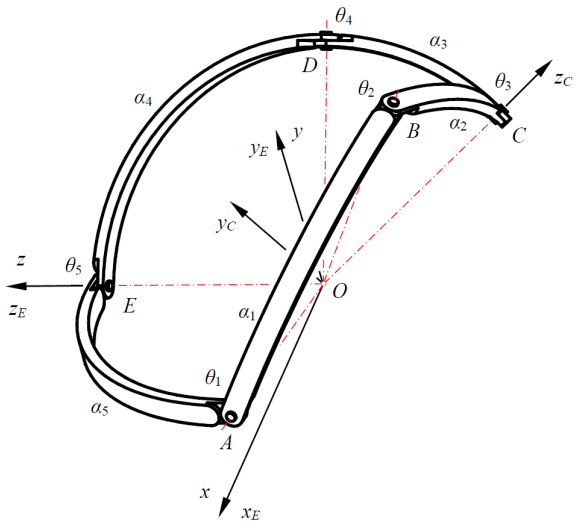


Figure 4.1: Example of a Spherical Five Bar Linkage.

The skeleton of the spherical five-bar linkage used for the palm, is shown in Fig. 4.1. We assume that the radius of the five-bar linkage is $r = 1$ so that the length of each link will be expressed by a value in radiants. This conserves generality since every spherical mechanism

can be projected onto the surface of a unit sphere. Every similar mechanism geometry can thus be adjusted to allow performing the calculations as described in this dissertation.

The mechanism consists of five spherical links with their lengths denoted as a_1 to a_5. The links are joined by five revolute joints with their angles denoted as θ_1 to θ_5. The rotation axes of all the revolute joints intersect at point O shown in Fig. 4.1. θ_1 is the crank input joint angle. θ_2 is the crank-coupler joint angle. θ_3 is the coupler joint angle. θ_4 is the coupler-rocker joint angle. θ_5 is the rocker input joint angle. In the linkage, the joints are numbered such that when all links lie on the same plane, the first joint on each link in a counter-clockwise direction shares the same link number. Each link is assigned a reference frame denoted as F_A to F_E. The Z_i-axis (i stands for A, B, C, D and E) of each frame originates at the spherical centre O and pass through the pivot axis of the joint with the same link number. The X_i-axis is on the plane formed by the two link joints and the spherical centre O and its positive direction is on the half plane that contains the second joint. The Y_i-axis is determined by the right hand rule. If the mechanism was to lie on a table, the links would be laid out in a counter-clockwise direction. Further, a global(reference) coordinate system $\{x, y, z\}$ is established with is origin located at point O. z-axis aligned with axis of joint E, and its y-axis directed along $Z_E \times Z_A$.

The method presented focuses on the cosine law for spherical triangles. The spherical mechanism that is the palm of the hand is decomposed into triangles. Their angles are then combined to compute the direct and inverse kinematics of the palm. This method is dealing with spherical five-bar kinematic chains of mobility $M = 2$ and the following characteristics:

$$0 < \alpha_i < \pi$$

where α_i with $i = 1, 2, 3, 4, 5$ are the angles of each link.

4.3 Spherical Trigonometry For The Metamorphic Palm

The spherical five-bar mechanism shown in Fig. 4.1 is used. It can be shown that each of the three output joints can be computed by an appropriate spherical triangle segmentation of the mechanism followed by application of the the spherical cosine law, as found in [5], with an appropriate adaptation of its terms as

$$\cos\theta_c = \frac{\cos\alpha_c - \cos\alpha_a \ \cos\alpha_b}{\sin\alpha_a \ \sin\alpha_b} \tag{4.1}$$

where, angles α_a and α_b correspond to the links forming the joint under consideration. Angle α_c corresponds to a third, virtual, link that completes the spherical triangle and angle θ_c is the (dihedral) joint angle being calculated, as is shown in Fig. 4.2.

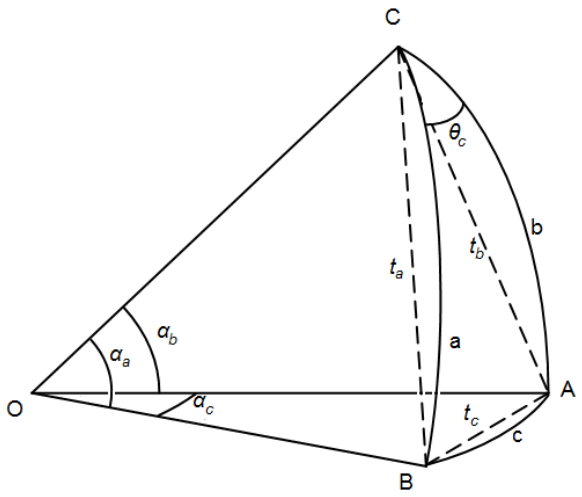


Figure 4.2: A Spherical Triangle.

The lengths of the adjacent links are known by the geometry of the spherical mechanism. The length of the hypothetical link can be computed by applying Eq. (4.2) on the chord joining the other ends of the two links under consideration. The distance between those ends

can be computed by the geometry and configuration of the mechanism.

$$\alpha_c = \cos^{-1}\left(1 - \frac{t_c^2}{2}\right) \tag{4.2}$$

The computation of joint-coupler angle θ_3 when the input joints are θ_1 and θ_5 is a direct application of Eq. (4.2) and Eq. (4.1). This can be seen in Eqs. (4.11), (5.7), (6.3), and (6.6). However, computing of crank-coupler joint-angle θ_2 and rocker-coupler joint angle θ_4 involves dividing the joints into more triangles and then combining them.

4.4 Position Analysis and Joint Axis

So as to construct the formulae based on Eq. (4.2), the coordinates of the points A, B, C, D and E as well as joint angles θ_2, θ_3 and θ_4 should be computed first. We start by assuming that point E is $\boldsymbol{p}_e = [0, 0, 1]$. Then the coordinates of points A, B and D are computed and expressed by the input joint angles θ_5 and θ_1. Then, angle θ_3 can be computed by applying the cosine law for spherical triangles on the triangle $\triangle BCD$. Computing rocker-coupler joint angle θ_4 can be done by combining dihedral angles θ_{eda}, θ_{adb}, θ_{bdc} and subtracting them from π. Crank-coupler joint Angle θ_2 can be computed in a similar way, by combining θ_{abe}, θ_{ebd}, θ_{dbc} and subtracting them from π. This indicates that the chords t_{bd}, t_{be} and t_{ad} have to be computed.

Referring to Fig. 4.1, the coordinates for points A, B, D and E can be computed by performing the rotations described as follows.

$$\boldsymbol{p}_a = R_y(\alpha_5)\, \boldsymbol{k} \tag{4.3}$$

$$\boldsymbol{p}_b = R_y(\alpha_5)\, R_z(\theta_1)\, R_y(\alpha_1)\, \boldsymbol{k} \tag{4.4}$$

$$\boldsymbol{p}_d = R_z(-\theta_5)\, R_y(-\alpha_4)\, \boldsymbol{k} \tag{4.5}$$

$$\boldsymbol{p}_e = \boldsymbol{k} \tag{4.6}$$

where, $\boldsymbol{k} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$. $R_y(\alpha_i)$ and $R_z(\theta_i)$ are standard rotation matrices as follows.

$$R_y(\alpha_i) = \begin{bmatrix} \cos\alpha_i & 0 & \sin\alpha_i \\ 0 & 1 & 0 \\ -\sin\alpha_i & 0 & \cos\alpha_i \end{bmatrix} \tag{4.7}$$

$$R_z(\theta_i) = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 \\ \sin\theta_i & \cos\theta_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.8}$$

4.5 Position Analysis for Cords of Spherical Links

Next, the chords t_{bd}, t_{be}, t_{ad} and the arc angles α_{bd}, α_{be} and α_{ad} can be calculated as follows based on Eq. (4.2).

Positions of $\boldsymbol{p}_b$ and $\boldsymbol{p}_d$ in Eqs. (4.4) and (4.5) yields chord t_{bd} as

$$\boldsymbol{p}_{bd} = \boldsymbol{p}_b - \boldsymbol{p}_d \tag{4.9}$$

$$t_{bd} = \sqrt{\boldsymbol{p}_{bd}^T\, \boldsymbol{p}_{bd}} \tag{4.10}$$

Substituting Eq. (4.10) into Eq. (4.2) results in

$$\alpha_{bd} = \cos^{-1}\left(1 - \frac{{t_{bd}}^2}{2}\right) \tag{4.11}$$

Similarly, positions of $\boldsymbol{p}_b$ and $\boldsymbol{p}_e$ in Eqs. (4.4) and (4.6) lead to chord t_{be} as

$$\boldsymbol{p}_{be} = \boldsymbol{p}_b - \boldsymbol{p}_e \tag{4.12}$$

$$t_{be} = \sqrt{\boldsymbol{p}_{be}^T \, \boldsymbol{p}_{be}} \tag{4.13}$$

Substituting Eq. (4.13) into Eq. (4.2) brings

$$\alpha_{be} = \cos^{-1}\left(1 - \frac{{t_{be}}^2}{2}\right) \tag{4.14}$$

Further, positions of $\boldsymbol{p}_a$ and $\boldsymbol{p}_d$ in Eqs. (4.3) and (4.5) yield chord t_{ad} as

$$\boldsymbol{p}_{ad} = \boldsymbol{p}_a - \boldsymbol{p}_d \tag{4.15}$$

$$t_{ad} = \sqrt{\boldsymbol{p}_{ad}^T \, \boldsymbol{p}_{ad}} \tag{4.16}$$

Substituting Eq. (4.16) into Eq. (4.2) results in

$$\alpha_{ad} = \cos^{-1}\left(1 - \frac{{t_{ad}}^2}{2}\right) \tag{4.17}$$

4.6 Spherical Segregation

The pivots of the spherical five-bar linkage, form a number of spherical triangles. With the values of arc angles α_{bd}, α_{be} and α_{ad} obtained above, referring to Fig. 4.1, dihedral angles θ_{eda}, θ_{ead}, θ_{aeb}, θ_{abe}, θ_{adb}, θ_{ebd}, θ_{dbc}, θ_{bdc} and θ_{bcd}, which are directly related to the joint angles θ_2, θ_3 and θ_4, can be formulated.

First, the cosine law for spherical triangles is applied to every triangle under consideration. If the magnitude of the cosine of any of the triangle angles evaluates as larger than one, then

the configuration is unsolvable. This would be either due to a singularity or the length of the links of the mechanism not being large enough. Next, the sine law for spherical triangles is applied. This allows solving for angles greater than π. Ultimately, the arctangent is used to derive the angle values. It is important to note that when implemented on a computer, the function Atan2 should be used, which provides the correct result when the angle is $k\pi, k \in \mathbb{N}^+$.

To prepare to solve for the rocker-coupler angle θ_{edc} and crank-coupler angle θ_{abc}, dihedral angles θ_{ead} and θ_{aeb} first need to be computed.

$$\theta_{ead} = Atan2\left(\sin\alpha_4 \frac{\sin\theta_5}{\sin\alpha_{ad}}, \frac{\cos\alpha_4 - \cos\alpha_5\ \cos\alpha_{ad}}{\sin\alpha_5\ \sin\alpha_{ad}}\right) \tag{4.18}$$

$$\theta_{aeb} = Atan2\left(\sin\alpha_1 \frac{\sin\theta_1}{\sin\alpha_{be}}, \frac{\cos\alpha_1 - \cos\alpha_5\ \cos\alpha_{be}}{\sin\alpha_5\ \sin\alpha_{be}}\right) \tag{4.19}$$

Next, rocker-coupler angle θ_{edc} segments are computed.

$$\begin{cases} \theta_{eda} = Atan2\left(\sin\alpha_5 \frac{\sin\theta_5}{\sin\alpha_{ad}}, \frac{\cos\alpha_5 - \cos\alpha_4\ \cos\alpha_{ad}}{\sin\alpha_4\ \sin\alpha_{ad}}\right) & (4.20) \\ \theta_{adb} = Atan2\left(\sin\alpha_1 \frac{\sin\left(\theta_1 - (-\theta_{ead})\right)}{\sin\alpha_{bd}}, \frac{\cos\alpha_1 - \cos\alpha_{ad}\ \cos\alpha_{bd}}{\sin\alpha_{ad}\ \sin\alpha_{bd}}\right) & (4.21) \\ \theta_{bdc} = \cos^{-1}\left(\frac{\cos\alpha_2 - \cos\alpha_3\ \cos\alpha_{bd}}{\sin\alpha_3\ \sin\alpha_{bd}}\right) & (4.22) \end{cases}$$

Crank-coupler angle θ_{abc} segments are then computed.

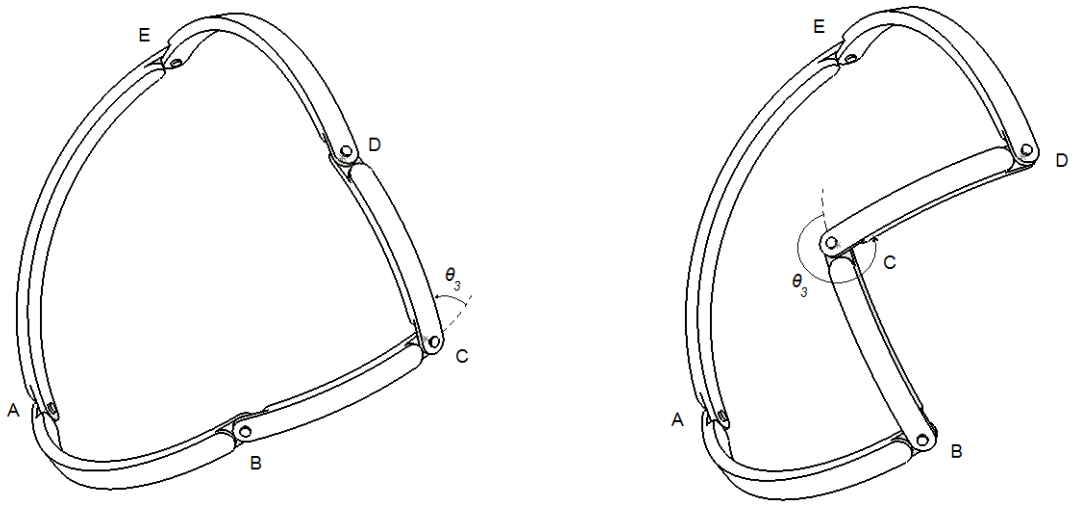
$$\begin{cases} \theta_{abe} = Atan2\left(\sin\alpha_5 \frac{\sin\theta_1}{\sin\alpha_{be}}, \frac{\cos\alpha_5 - \cos\alpha_1\ \cos\alpha_{be}}{\sin\alpha_1\ \sin\alpha_{be}}\right) & (4.23) \\ \theta_{ebd} = Atan2\left(\sin\alpha_4 \frac{\sin\left(\theta_5 - (-\theta_{aeb})\right)}{\sin\alpha_{bd}}, \frac{\cos\alpha_4 - \cos\alpha_{be}\ \cos\alpha_{bd}}{\sin\alpha_{be}\ \sin\alpha_{bd}}\right) & (4.24) \\ \theta_{dbc} = \cos^{-1}\left(\frac{\cos\alpha_3 - \cos\alpha_2\ \cos\alpha_{bd}}{\sin\alpha_2\ \sin\alpha_{bd}}\right) & (4.25) \end{cases}$$

Finally, coupler dihedral angle θ_{bcd} is computed.

$$\theta_{bcd} = \cos^{-1}\left(\frac{\cos\alpha_{bd} - \cos\alpha_2\ \cos\alpha_3}{\sin\alpha_2\ \sin\alpha_3}\right) \tag{4.26}$$

4.7 Reflex and Non-Reflex Joint-Coupler Configurations

The coupler links can have two distinct configurations, as shown in Fig. 4.3. The two configurations are differentiated with respect to the value of the coupler joint angle being greater or less than π. In the first case, coupler joint angle θ_3 is less than π and is thus entitled as the joint-coupler non-reflex configuration, as show in Fig. 4.3a. In the case where the joint-coupler angle θ_3 is larger than π, the mechanism is in the reflex joint-coupler configuration, as shown in Fig. 4.3b.



(a) CAD Drawing, Joint-Coupler Non-Reflex Configuration. (b) CAD Drawing, Joint-Coupler Reflex Configuration.

Figure 4.3: Joint-Coupler Non-Reflex and Reflex Configurations.

To compute the output joint angles in the reflex configuration, the triangle angles are combined as follows.

$$\theta_{4r} = \pi - (\theta_{eda} + \theta_{adb} - \theta_{bdc}) = f_{4r}(\theta_1, \theta_5) \tag{4.27}$$

$$\theta_{2r} = \pi - (\theta_{abe} + \theta_{ebd} - \theta_{dbc}) = f_{2r}(\theta_1, \theta_5) \tag{4.28}$$

$$\theta_{3r} = \pi + \theta_{bcd} = f_{3r}(\theta_1, \theta_5) \tag{4.29}$$

In the non-reflex joint-coupler configuration it is

$$\theta_{4nr} = \pi - (\theta_{eda} + \theta_{adb} + \theta_{bdc}) = f_{4nr}(\theta_1, \theta_5) \tag{4.30}$$

$$\theta_{2nr} = \pi - (\theta_{abe} + \theta_{ebd} + \theta_{dbc}) = f_{2nr}(\theta_1, \theta_5) \tag{4.31}$$

$$\theta_{3nr} = \pi - \theta_{bcd} = f_{3nr}(\theta_1, \theta_5) \tag{4.32}$$

where the number in the subscript denotes the angle index number. The letter in the subscript denotes the reflex or non-reflex joint-coupler configuration.

To obtain the inverse kinematics for the palm, the same set of equations is used with a simple cyclic permutation of the arc and dihedral joint angles, as long as the given dihedral joint angles are adjacent. For example, if values for θ_3 and θ_4 are given, the following permutations are necessary.

$$\begin{pmatrix} \theta_3 & \theta_4 & \theta_5 & \theta_1 & \theta_2 \\ \theta_5 & \theta_1 & \theta_2 & \theta_3 & \theta_4 \end{pmatrix}$$

and

$$\begin{pmatrix} a_3 & a_4 & a_5 & a_1 & a_2 \\ a_5 & a_1 & a_2 & a_3 & a_4 \end{pmatrix}$$

where θ_i are the dihedral joint angles of the mechanism. a_i are the arc lengths corresponding to each link of the mechanism.

4.8 Position Analysis of the Joint-Coupler of the Palm

Finally, point C can be located by either following the E, D, C path, as seen in Eq. (4.33), or the E, A, B, C path as in Eq. (4.34). The first way is the least computationally intensive, as it involves fewer terms.

$$\boldsymbol{p}_c = R_y(-\alpha_3)\, R_z(-\theta_4)\, R_y(-\alpha_4)\, R_z(-\theta_5)\, \boldsymbol{k} \tag{4.33}$$

$$\boldsymbol{p}_c = R_y(\alpha_2)\, R_z(\theta_2)\, R_y(\alpha_1)\, R_z(\theta_1)\, R_y(\alpha_5)\, \boldsymbol{k} \tag{4.34}$$

4.9 Conclusions

In this chapter a method for deriving the kinematics of spherical five-bar mechanisms based on simple math was presented. The method is based on segregating the mechanism into spherical triangles, solving the triangles by means of spherical trigonometry and then combining the partial solutions to form a complete solution for the mechanism. The same set of equations can be used for both forward and inverse kinematics, just by a simple cyclic permutation of the joint and link numbers.

It was shown that by using spherical trigonometry, with the intrinsic constraint of all links and joint angles being on the surface of a sphere, requires a small number of steps to derive a solution. The resultant equations are easy to formulate and follow. This simplicity has the added benefit, as will be shown in chapter 5, of identifying singular configurations just by inspection.

The presented method allows differentiation of the joint-coupler reflex and non-reflex configurations, something that was either not possible or practical with previous formulation methods. This enables accurate solving of the joint angles of the palm mechanism in both

configurations. The math is simple to implement in a computer program. They only require computing of a small number of basic trigonometric quantities and only a few additions, multiplications and divisions. All of which can be executed fast in a modern embedded control system. Because of the simplicity of the solution, the possibility of human error is also reduced compared to methods based on D-H parameters or quaternions.

Chapter 5

Singularity Analysis of the Metamorphic Palm

5.1 Introduction

Singular configurations are present in most mechanisms. More so in metamorphic mechanisms who employ singular configurations to transition from one mobility phase to another. It is important to study and understand all the singularities of each metamorphic mechanism in order to be able to control the mechanism and take advantage of the singular configuration. This chapter uses the spherical trigonometry-constrained kinematics as shown in chapter 4 as a basis for identifying the singular configurations of the metamorphic palm. Then the work proceeds in identifying the singularities into the ones where control is lost, and into the ones where the palm is still controllable. Singularity avoidance-based design criteria are provided for the former case. Solutions are provided for the latter case.

5.2 Spherical Trigonometry-Constrained Kinematics

The spherical trigonometry-constrained kinematics presented in chapter 4 are used to perform the singularity analysis of the metamorphic palm. The part of the results most relevant to the singularity analysis is reiterated, as follows.

$$\begin{cases} \theta_{eda} = Atan2\left(\sin\alpha_5 \dfrac{\sin\theta_5}{\sin\alpha_{ad}}, \dfrac{\cos\alpha_5 - \cos\alpha_4\ \cos\alpha_{ad}}{\sin\alpha_4\ \sin\alpha_{ad}}\right) & (5.1) \\ \theta_{adb} = Atan2\left(\sin\alpha_1 \dfrac{\sin\left(\theta_1 - (-\theta_{ead})\right)}{\sin\alpha_{bd}}, \dfrac{\cos\alpha_1 - \cos\alpha_{ad}\ \cos\alpha_{bd}}{\sin\alpha_{ad}\ \sin\alpha_{bd}}\right) & (5.2) \\ \theta_{bdc} = \cos^{-1}\left(\dfrac{\cos\alpha_2 - \cos\alpha_3\ \cos\alpha_{bd}}{\sin\alpha_3\ \sin\alpha_{bd}}\right) & (5.3) \end{cases}$$

Crank-coupler angle θ_{abc} segments are then computed.

$$\begin{cases} \theta_{abe} = Atan2\left(\sin\alpha_5 \dfrac{\sin\theta_1}{\sin\alpha_{be}}, \dfrac{\cos\alpha_5 - \cos\alpha_1\ \cos\alpha_{be}}{\sin\alpha_1\ \sin\alpha_{be}}\right) & (5.4) \\ \theta_{ebd} = Atan2\left(\sin\alpha_4 \dfrac{\sin\left(\theta_5 - (-\theta_{aeb})\right)}{\sin\alpha_{bd}}, \dfrac{\cos\alpha_4 - \cos\alpha_{be}\ \cos\alpha_{bd}}{\sin\alpha_{be}\ \sin\alpha_{bd}}\right) & (5.5) \\ \theta_{dbc} = \cos^{-1}\left(\dfrac{\cos\alpha_3 - \cos\alpha_2\ \cos\alpha_{bd}}{\sin\alpha_2\ \sin\alpha_{bd}}\right) & (5.6) \end{cases}$$

Finally, coupler dihedral angle θ_{bcd} is computed.

$$\theta_{bcd} = \cos^{-1}\left(\frac{\cos\alpha_{bd} - \cos\alpha_2\ \cos\alpha_3}{\sin\alpha_2\ \sin\alpha_3}\right) \quad (5.7)$$

5.3 From Kinematics to Singularities

When the denominators of the above equations become zero, these kinematic equations do not have a solution. As such, type I singularities occur. This is when

$$\sin\alpha_4 \ \sin\alpha_{ad} = 0 \tag{5.8}$$

$$\sin\alpha_{ad} \ \sin\alpha_{bd} = 0 \tag{5.9}$$

$$\sin\alpha_3 \ \sin\alpha_{bd} = 0 \tag{5.10}$$

$$\sin\alpha_1 \ \sin\alpha_{be} = 0 \tag{5.11}$$

$$\sin\alpha_{be} \ \sin\alpha_{bd} = 0 \tag{5.12}$$

$$\sin\alpha_2 \ \sin\alpha_{bd} = 0 \tag{5.13}$$

$$\sin\alpha_2 \ \sin\alpha_3 = 0 \tag{5.14}$$

Which in turn can be broken down into the trivial

$$\sin\alpha_4 = 0 \tag{5.15}$$

$$\sin\alpha_3 = 0 \tag{5.16}$$

$$\sin\alpha_1 = 0 \tag{5.17}$$

$$\sin\alpha_2 = 0 \tag{5.18}$$

$$\sin\alpha_3 = 0 \tag{5.19}$$

and non trivial

$$\sin\alpha_{ad} = 0 \tag{5.20}$$

$$\sin\alpha_{bd} = 0 \tag{5.21}$$

$$\sin\alpha_{be} = 0 \tag{5.22}$$

which can be further refined into the following three conditions

$$t_{AD} = 0, 2 \tag{5.23}$$

$$t_{BE} = 0, 2 \tag{5.24}$$

$$t_{BD} = 0, 2 \tag{5.25}$$

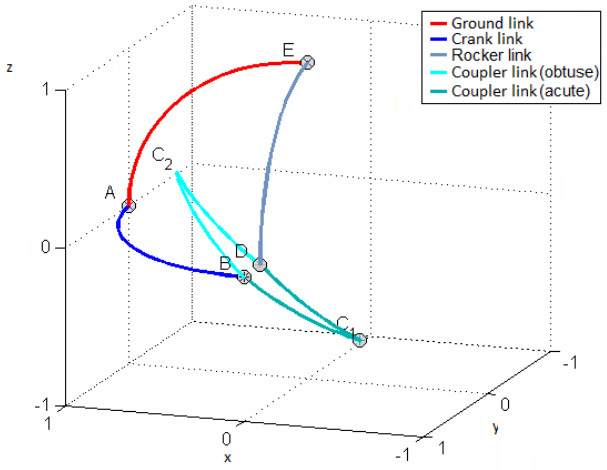
Thus, the mechanism is analysed. Singularity avoidance-based criteria are presented, and a solution for the singular and controllable case is identified.

5.4 Singular Configurations of the Metamorphic Palm

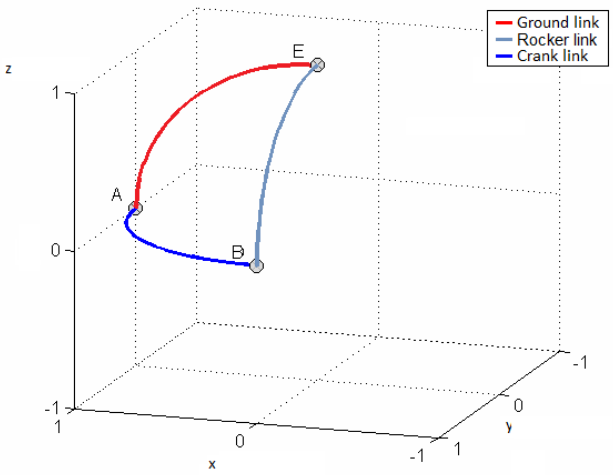
The singular configurations become apparent when inspecting the outcome of applying the cosine law to the spherical triangles, as indicated in Section 5.2. A singularity occurs when two joints axes are collinear. However, loss of control happens only when both collinear joints are output joints. If the two joints happen to be one output and one input joint, then the mechanism can still be controllable as a one DOF mechanism. Equations (4.18) to (5.6) reveal these exact singularities of the linkage.

Fig. 5.1b illustrates this singular configuration by using a spherical mechanism with link angles $\alpha_4 = \alpha_5 = \alpha_1 = \frac{\pi}{2}$ and $\alpha_2 = \alpha_3 = \frac{\pi}{4}$ and joint angles $\theta_1 = 90$ and $\theta_5 = 90$. Link AE is the ground link. Link AB is the crank and link DE is the rocker. Links B_iC_i and C_iD_i are the two possible configurations for the couplers.

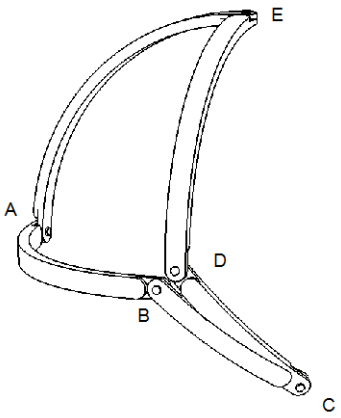
At the singular configuration, points B and D overlap, joint angle $\theta_3 = k\,\pi\,, k \in N^+$ and joint angles θ_2 and θ_4 are undefined.



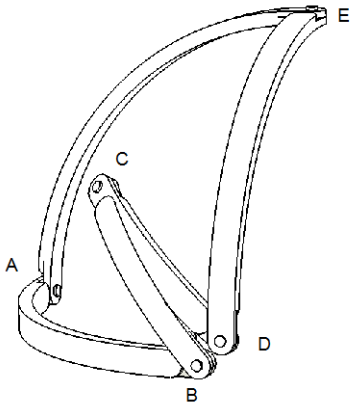
(a) Approaching a Singularity.



(b) Singular Configuration.



(c) Approaching a Singularity, CAD Drawing, Coupler-Non-Reflex-Angle.



(d) Approaching a Singularity, CAD Drawing, Coupler-Reflex-Angle.

Figure 5.1: Singular Configuration.

5.5 Singularity Avoidance Based Design Criteria

In particular, to avoid a singularity, Eqn. 5.20 to 5.25 must not be realised. As such, all of the following conditions must be met.

$$\sin \alpha_{ad} \neq 0 \tag{5.26}$$

$$\sin \alpha_{be} \neq 0 \tag{5.27}$$

$$\sin \alpha_{bd} \neq 0 \tag{5.28}$$

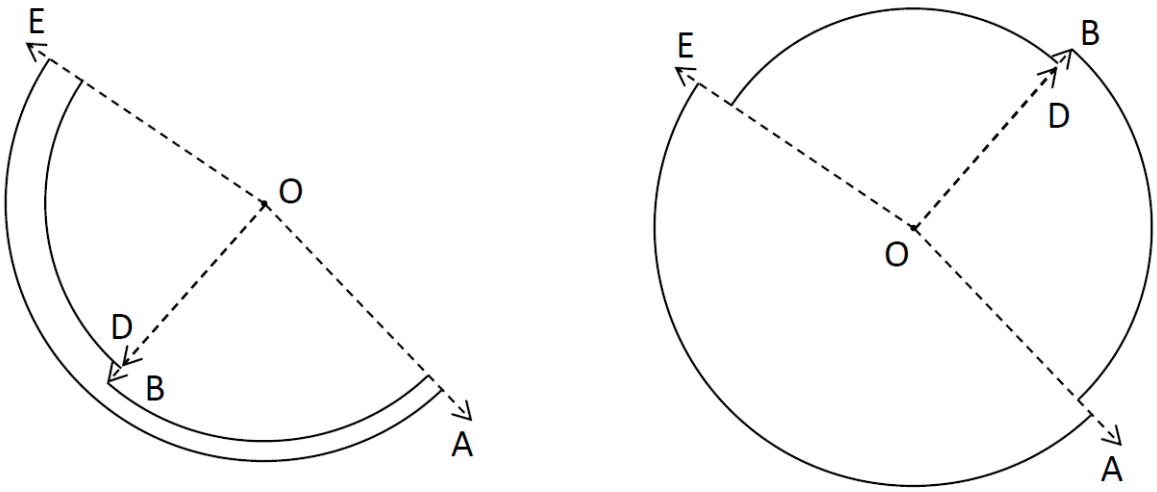
which can be further refined to the next three conditions

$$t_{AD} \neq 0, 2 \tag{5.29}$$

$$t_{BE} \neq 0, 2 \tag{5.30}$$

$$t_{BD} \neq 0, 2 \tag{5.31}$$

However loss of control only occurs in the case where $t_{BD} = 0 or 2$. There are four distinct cases where the mechanism is singular and it is impossible to control all output joints.

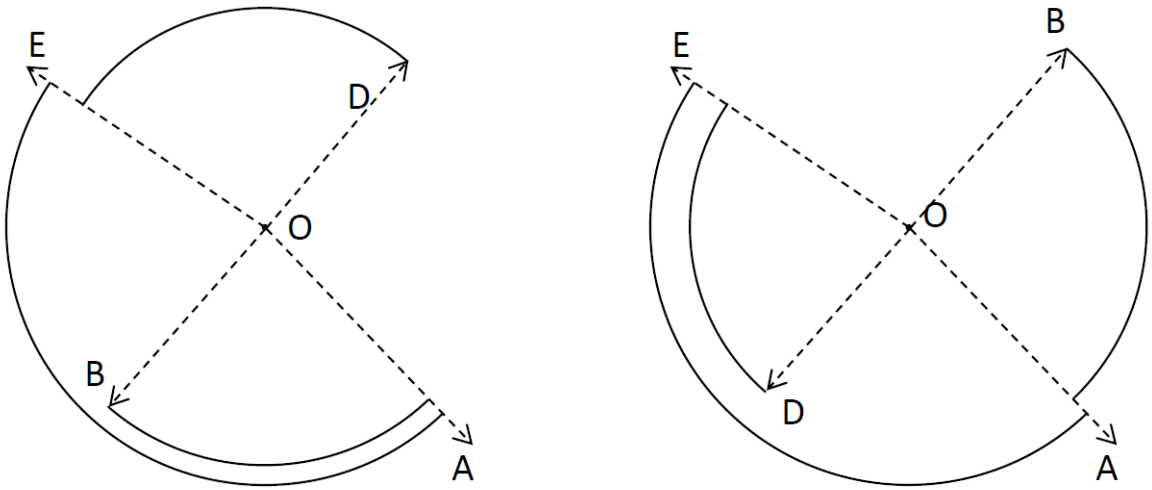


(a) Singularity No. 1.

(b) Singularity No. 2.

Figure 5.2: Singular Configurations No. 1 and 2.

The first two cases the mechanism can become uncontrollable are shown in Fig. 5.2. In both cases it is $t_{BD} = 0$. In the first case, as shown in Fig. 5.2a, the crank and rocker are rotated towards each other and lie on the plane and within the arc defined by the line segments OE and OA. As long as the combined lengths of the crank and coupler are larger or equal to the length of the ground link, it is possible for axes z_2 and z_4 to become collinear. Fig. 5.2b shows the other possible configuration where it is also $t_{BD} = 0$. In this case the rocker and coupler are also rotated towards each other. They lie on the plane and outside of the arc again defined by the line segments OA and OE. In both cases, it must be $\alpha_2 = \alpha_3$ or it will be impossible for points B and D to coincide.



(a) Singularity No. 3.

(b) Singularity No. 4.

Figure 5.3: Singular Configurations No. 3 and 4.

The other two cases where the mechanism can become uncontrollable are shown in Fig. 5.3. In both cases it is $t_{BD} = 2$. In the third case, as shown in Fig. 5.3a, the crank and rocker are rotated away from each other and must be coplanar so they lie on the plane defined by the line segments OE and OA. Referencing the arc defined by those two line segments, the crank

lies within the arc, and the rocker lies outside of it. If the sum of the angle of the arc of the rocker and ground link, minus the angle of the arc of the crank is equal to π, then z_2 and z_4 can become collinear.

Figure 5.3b shows the other possible configuration where it is also $t_{BD} = 2$. The crank and rocker are again rotated away from each other, are coplanar between them and with the the plane defined by the line segments OE and OA. Referencing the arc defined by those two line segments, the rocker lies within it and the crank lies outside of it. If the sum of the angle of the arc of the crank and ground link minus the angle of the arc of the rocker is equal to π, then z_2 and z_4 can become collinear. In both cases the points B and D can coincide regardless whether $\alpha_2 = \alpha_3$ or $\alpha_2 \neq \alpha_3$.

To avoid singularities it must be:

$$\begin{cases} \alpha_4 + \alpha_5 - \alpha_1 \neq \pi & (5.32) \\ \alpha_1 + \alpha_5 - \alpha_4 \neq \pi & (5.33) \end{cases}$$

And if $\alpha_2 = \alpha_3$ then it must also be:

$$\begin{cases} \alpha_1 + \alpha_4 < \alpha_5 & (5.34) \\ \alpha_1 + \alpha_4 + \alpha_5 < 2\pi & (5.35) \end{cases}$$

5.6 The 1-DOF Singular but Solvable Case

While loss of control can occur if the axes of joints θ_2 and θ_4 coincide, this is not the case when any of these two output joints becomes collinear with one of the two input joints. In this special case, a slight change in the formulation of the kinematics is necessary.

An example of a spherical five bar mechanism reduced to a 1-DOF mechanism is shown in Fig. 5.4. No collision detection is performed in this case. There is no loss of generality since

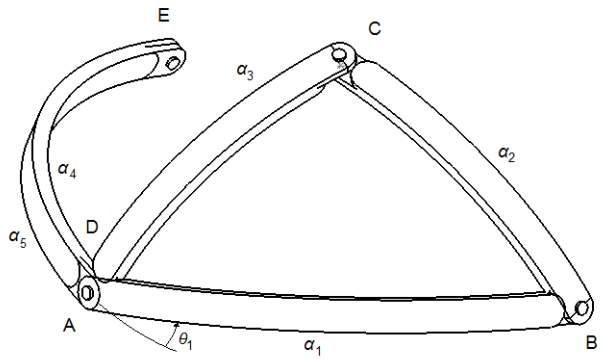


Figure 5.4: Singular and Controllable Case.

the important feature is the axes corresponding to points A and D becoming collinear. Links α_5 and α_4 are oriented in order to overlap. Thus points A and D coincide and the mechanism is becoming singular. Joint θ_1 that corresponds to point A is an input joint. In a practical mechanism it is connected to an actuator. Joint θ_4 which corresponds to point D is an output joint and not controlled directly. In this special case, the mechanism is controllable as follows.

In the case where the two collinear axes are one input and one output axis, the mechanism is still controllable but reduces to one DOF. In such a case, the following change happens in the kinematics equations 6.5, 6.2, 6.4, and 6.1

$$\theta_{abe} + \theta_{ebd} = \begin{cases} \pi - \theta_5 & \text{if } t_{AD} = 0 \\ \theta_5 & \text{if } t_{AD} = 2 \end{cases} \tag{5.36}$$

$$\theta_{eda} + \theta_{adb} = \begin{cases} \pi - \theta_1 & \text{if } t_{BE} = 0 \\ \theta_1 & \text{if } t_{BE} = 2 \end{cases} \tag{5.37}$$

5.7 Conclusions

In this chapter, singularity analysis of the metamorphic palm was performed. It is important to identify the singular configurations of the metamorphic palm in order to either avoid them during design and/or planning, or still be able to control the mechanism if singular.

The very nature of spherical trigonometry-constrained kinematics greatly simplifies the singularity analysis of the metamorphic palm. Simple trigonometry yields elegant results. Through spherical trigonometry, singularities become obvious by inspecting the equations of the angles of each spherical triangle the palm is decomposed into. Just by equating the denominators of all fractions as they appear in the kinematic equations, is enough to identify all singular configurations of the mechanism.

All singular configurations were identified and analysed. In two cases the mechanism is singular and uncontrollable. In four singular cases the mechanism is singular but can still be controlled. A set of singularity-avoidance based design criteria was developed, based on the singular and uncontrollable cases. For the singular but controllable cases, the required changes to the spherical trigonometry-based kinematics equations are presented so a solution is shown.

Chapter 6

Simulation in the Joint Space and Workspace

6.1 Introduction

The metamorphic palm was designed to form a basis for metamorphic hands. Its purpose is to enhance the workspace and dexterity of a robotic hand by being used in place of a fixed palm. By changing the palm configuration, the operation planes of the fingers can be oriented in interesting ways. This allows for more complex hand poses that would otherwise be possible.

It is important to perform simulations of different metamorphic palm geometries in the joint space and in the workspace. Such simulations could be used for analysis of the palm and to provide examples of how the palm enhances the workspace and joint space of the hand. Existing methods do not differentiate the joint-coupler reflex and non-reflex configurations. This is very important when simulating a metamorphic palm-based robotic hand.

By using the spherical trigonometry-constrained kinematics and singularity analysis results

of chapters 4 and 5, we are now able to construct computer simulation models of the palm and hand in the joint space and in the workspace. These models are able to differentiate the joint-coupler reflex and non-reflex configurations.

6.2 Spherical Trigonometry constrained Kinematics

In order to perform joint space and workspace simulation, the spherical trigonometry-constrained kinematics are used. They are repeated below as follows.

In the joint-coupler reflex configuration it is

$$\theta_{4r} = \pi - (\theta_{eda} + \theta_{adb} - \theta_{bdc}) = f_{4r}(\theta_1, \theta_5) \tag{6.1}$$

$$\theta_{2r} = \pi - (\theta_{abe} + \theta_{ebd} - \theta_{dbc}) = f_{2r}(\theta_1, \theta_5) \tag{6.2}$$

$$\theta_{3r} = \pi + \theta_{bcd} = f_{3r}(\theta_1, \theta_5) \tag{6.3}$$

In the joint-coupler non-reflex configuration it is

$$\theta_{4nr} = \pi - (\theta_{eda} + \theta_{adb} + \theta_{bdc}) = f_{4nr}(\theta_1, \theta_5) \tag{6.4}$$

$$\theta_{2nr} = \pi - (\theta_{abe} + \theta_{ebd} + \theta_{dbc}) = f_{2nr}(\theta_1, \theta_5) \tag{6.5}$$

$$\theta_{3nr} = \pi - \theta_{bcd} = f_{3nr}(\theta_1, \theta_5) \tag{6.6}$$

In the singular but controllable cases the following apply

$$\theta_{abe} + \theta_{ebd} = \begin{cases} \pi - \theta_5 & \text{if } t_{AD} = 0 \\ \theta_5 & \text{if } t_{AD} = 2 \end{cases} \tag{6.7}$$

$$\theta_{eda} + \theta_{adb} = \begin{cases} \pi - \theta_1 & \text{if } t_{BE} = 0 \\ \theta_1 & \text{if } t_{BE} = 2 \end{cases} \tag{6.8}$$

With the above results, it is possible to compute the configuration of the metamorphic palm given two input joint angles and the desired reflex or non-reflex configuration for the joint-coupler.

6.3 Kinematics for a Metamoprhic Palm-Based Robotic Hand

The position analysis equations for computing the coordinates of the fingers are used as described by Wei et al. [3]. These equations follow for reference purposes.

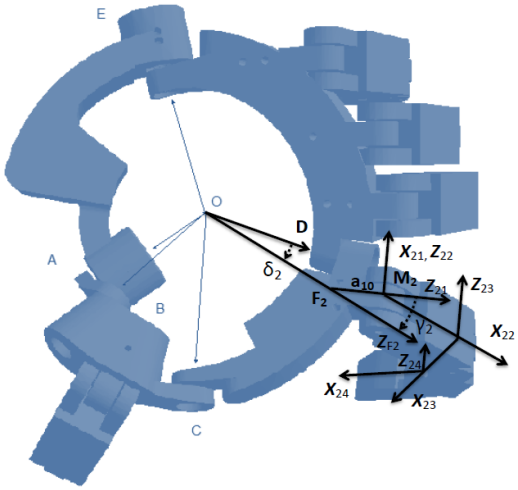


Figure 6.1: Metamorphic Hand Joint Axes.

First, each hand link needs a reference frame attached to it. The joint axes used for each finger are shown in Fig. 6.1. Axes Z_{Fi} point from the spherical centre to the start of the carpal link of each finger, at points F_i. Axes Z_{i1} start at the palm attachment and point to

the metacarpalphalangeal (MCP) joint of each finger, points M_i. Axes X_{i2} start from each MCP joint and point to the proximal interphalangeal joint (PIP) of each finger. Axes X_{i3} start at the PIP joints and point towards the distal interphalangeal joints (DIP). Finally, axes X_{i4} start at the DIP joints and run parallel to the distal phalanx of each finger.

First, angles θ_2, θ_3 and θ_4 from the spherical trigonometry-constrained kinematics are determined. Then, the points where the fingers attach to the palm, as depicted in Fig. 6.1, have to be determined.

Angles δ_1 to δ_5 determine points F_1 to F_5 where each proximal phalanges attach on the palm. Angles γ_2 to γ_5 are calculated and fixed so that each finger, except for the thumb, is parallel and co-linear with the arm. γ_1 is 0.

The values of the angles δ_1 to δ_5 are given by equations 6.9, 6.10, 6.11, 6.12 and 6.13.

$$\delta_1 = \alpha_2 \frac{1}{2} \tag{6.9}$$

$$\delta_2 = \alpha_3 \frac{1}{4} \tag{6.10}$$

$$\delta_3 = \alpha_4 \frac{1}{7} \tag{6.11}$$

$$\delta_4 = \alpha_4 \frac{3}{7} \tag{6.12}$$

$$\delta_5 = \alpha_4 \frac{5}{7} \tag{6.13}$$

The values of the angles γ_1 to γ_5 are given by equations 6.15, 6.15, 6.16, 6.17 and 6.18.

$$\gamma_1 = 0 \tag{6.14}$$

$$\gamma_2 = -\delta_2 - (\alpha_4 - \frac{\pi}{2}) \tag{6.15}$$

$$\gamma_3 = \delta_3 - (\alpha_4 - \frac{\pi}{2}) \tag{6.16}$$

$$\gamma_4 = \delta_4 - (\alpha_4 - \frac{\pi}{2}) \tag{6.17}$$

$$\gamma_5 = \delta_5 - (\alpha_4 - \frac{\pi}{2}) \tag{6.18}$$

In order to obtain the coordinates that describe the MCP joints, the homogeneous transform matrices to points F_i need to be determined.

$$R_{F_1} = R(y_5, \alpha_5)\ R(z_1, \theta_1)\ R(y_1, \alpha_1)\ R(z_2, \theta_2)\ R(y_2, \delta_1) \tag{6.19}$$

$$R_{F_2} = R(z_5, -\theta_5)\ R(y_4, -\alpha_4)\ R(z_4, -\theta_4)\ R(y_3, \delta_2) \tag{6.20}$$

$$R_{F_3} = R(z_5, -\theta_5)\ R(y_4, -(\alpha_4 - \delta_3)) \tag{6.21}$$

$$R_{F_4} = R(z_5, -\theta_5)\ R(y_4, -(\alpha_4 - \delta_4)) \tag{6.22}$$

$$R_{F_5} = R(z_5, -\theta_5)\ R(y_4, -(\alpha_4 - \delta_5)) \tag{6.23}$$

After obtaining the rotation matrices R_{F_i}, the coordinates of points F_i are easily obtained by eq. 6.24.

$$\mathbf{f_i} = R_{F_i}\ \mathbf{k} \tag{6.24}$$

Then, the homogeneous transform matrices for points F_i are given by eq. 6.25.

$$D_{F_i} = \begin{bmatrix} R_{F_i} & R_{F_i}\ \mathbf{k} \\ 0 & 1 \end{bmatrix} \tag{6.25}$$

Next, matrices D_{F_i} are multiplied by the homogeneous transform matrix D_{FM_i} given by eq. 6.26.

$$D_{FM_i} = \begin{bmatrix} \cos\gamma_i & 0 & -\sin\gamma_i & -a_{i0}\sin\gamma_i \\ 0 & 1 & 0 & 0 \\ \sin\gamma_i & 0 & \cos\gamma_i & a_{i0}\cos\gamma_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.26}$$

where a_{i0} are the distances from points F_i to M_i. The homogeneous transform matrices to points M_1 to M_5 are given by eq. 6.27 and eq. 6.28.

$$D_{M_1} = D_{F_1}\, D_{FM_1}\, D_{10} \tag{6.27}$$

$$D_{M_i} = D_{F_i}\, D_{FM_i} \tag{6.28}$$

where

$$D_{10} = \begin{bmatrix} \cos\theta_{10} & -\sin\theta_{10} & 0 & 0 \\ \sin\theta_{10} & \cos\theta_{10} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.29}$$

Finally, the coordinates for the points M_1 to M_5 can be computed from eq. 6.30.

$$\begin{bmatrix} \mathbf{r_{i0}} \\ 0 \end{bmatrix} = D_{M_i} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \tag{6.30}$$

Computing the finger joints is trivial as they are simple RRR serial mechanisms.

6.4 Hand Workspace Memory Storage Format

The workspace is generated by means of a nested for loop. All joints start with an angle in the home position, then advanced by a fixed step. If all points are stored in memory and plotted, this leads to a very large number of points to handle. Further, there is significant redundancy since many different configurations result in points very close to each other. If three nested "for" loops are used, if the step is set to 1 degree, then 4212400 points are generated.

To be able to store and retrieve the hand workspace easily, while eliminating redundant points, the workspace is stored in a three-dimensional array in computer memory. This procedure is also similar to the one found in [86]. The Cartesian space surrounding the hand is segmented into cells. This is done by surrounding the hand with a three dimensional array. As a result, in the case of Fig. 6.2 only 64000 points are stored and plotted. Further, the number of points is fixed and does not depend on the specifics, such as resolution, of the simulation algorithm.

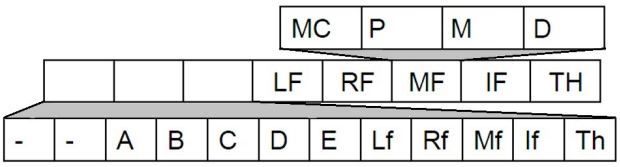


Figure 6.2: Workspace Point Memory Storage Format.

The structure of each cell of the array is shown in Fig. 6.2. Each cell contains a 32bit word. Each point of interest on the hand is assigned a single bit of the word. In particular, the least important half bytes are assigned to the thumb, index finger, middle finger, ring finger and little finger in that order. The bits in each half byte correspond to the ends of the metacarpal (MC), proximal (P), intermediate (I) and distal (D) phalanges for each finger. The twelve remaining bits of the word are assigned to the palm joints and the carpal metacarpal joints

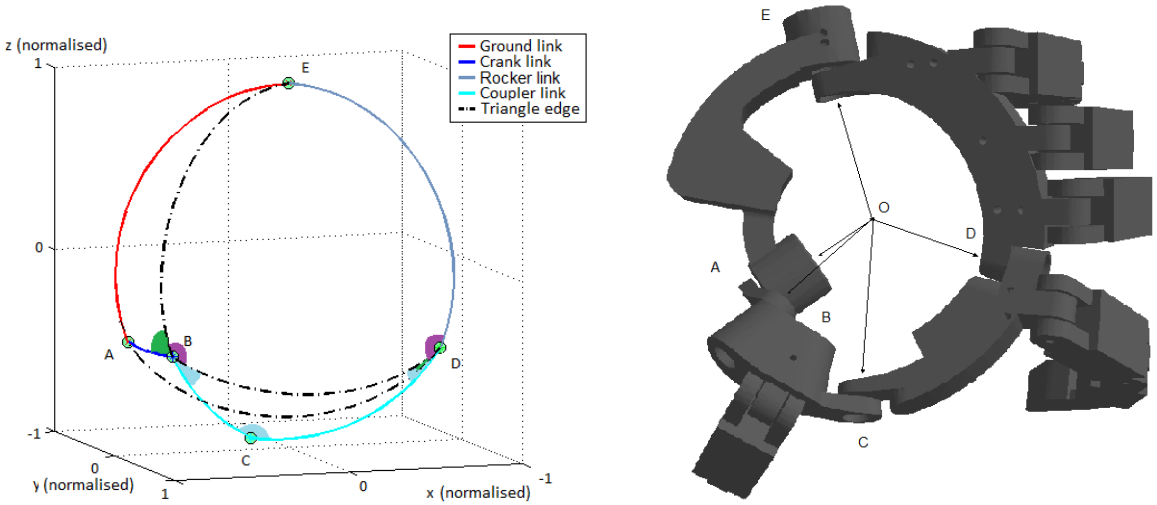
(CMC). The produced array can then be stored in computer memory for further processing, eliminating the need to regenerate it each time it is needed. This is very important since depending on the chosen cell size and available computer hardware, computation can take considerable time. This method also allows for a measure of workspace coverage. This can be accomplished by counting the number of cells that contain non-zero values and comparing them to the total number of cells.

6.5 Joint Space of the Palm of the KCL Metamorphic Hand.

The method presented in this chapter, is used to visualise the joint space of the KCL metamorphic hand. The joint space of the KCL Metamorphic hand is generated by varying the two input angles across all possible angle positions with a step of two degrees. Then all joint angles for the output angles when the palm is not in a non singular configuration are recorded.

The model of the metamorphic palm and the spherical triangles used to compute its kinematics is shown in Fig. 6.3a. The palm of the hand is projected on the surface of a unit sphere. This projection preserves generality since only angles are considered. The angles being combined are coloured for clarity. They are combined according to Eqs. (6.4) to (6.3) to derive θ_2, θ_3 and θ_4 for both elbow up and elbow down configurations.

Of particular interest are the joint-coupler angle θ_3 and the coupler-rocker joint angle θ_4. Figure 6.4 and Fig. 6.5 show the joint space for each joint is a manifold in a three dimensional space where the two dimensions are the input joint angles the third dimension is the angle of the joint studied. The values for each joint are shown for both the coupler joint reflex and non-reflex configurations. It can be seen when one of the two joints is at 0 degrees, the other



(a) Computer Visualisation of Metamorphic Palm Triangles. (b) Computer Rendering of the Metamorphic Hand.

Figure 6.3: Metamorphic Palm Triangles

input joint can only have the same value. This is the singular configuration where the palm of the hand is flat.

Figure 6.4 highlights the ability of the metamorphic palm to provide great flexibility in changing the relative orientations of the working planes of the index and thumb. It can be seen the coupler joint angle θ_3 can reach a great number of different configurations. The proposed method correctly distinguishes both reflex and non-reflex coupler joint configurations. All data points of θ_{3nr} are contained in the upper half of the graph corresponding to $0 < \theta_{3nr} < \pi$ and all data points of θ_{3r} are on the bottom half corresponding to $\pi < \theta_{3nr} < 2\pi$. Finally, the planes shown in grey are where $\theta_3 = 0$ and $\theta_3 = \pi$ and so the joint-coupler is singular.

The rocker-coupler joint angle θ_4 is limited compared to all other joints, as shown in Fig. 6.5. The planes in grey are when the joint is singular. It can be seen that joint θ_4 is singular only in the flat configuration. It is in fact the joint with the least possible attainable joint angles. This feature is desirable since collisions among fingers need to be avoided. In

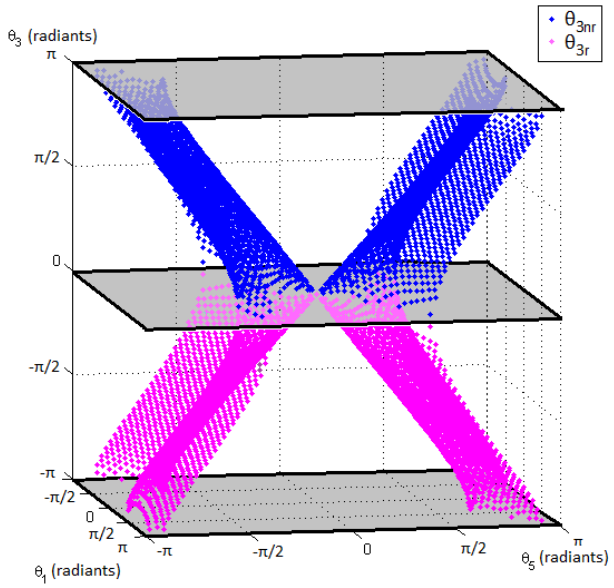


Figure 6.4: Joint Space of θ_3 for reflex and non-reflex configurations.

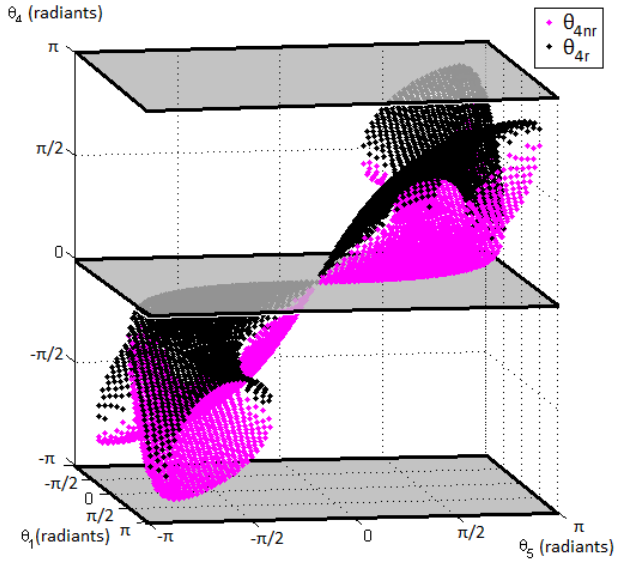


Figure 6.5: Joint Space of θ_4.

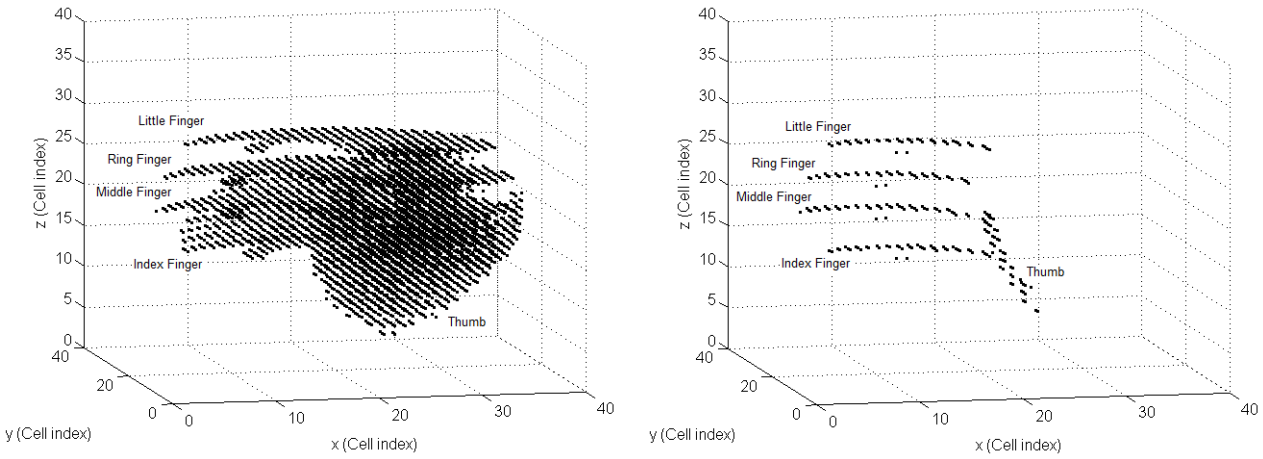
particular, it can be seen that rocker input joint angle θ_5 can change significantly before rocker-coupler joint angle θ_4 starts to deviate from 0 radiants. This reduces the risk of collision or significant change int the orientation of the index finger relative to the middle and little grasping fingers.

Since not all configurations are possible, when motion planning from one configuration to another, a path can be generated that lies on the surface of the joint space manifold for each joint. This path planning could be done by setting a starting point and a goal, and then using a path planning algorithm on the point cloud of the appropriate joint space. Once this is done, the path can be projected onto the actuated joint plane.

6.6 Simulation and Visualisation of the KCL Metamorphic Hand

Next, the workspace of the hand is computed. This is accomplished by using the spherical trigonometry-constrained kinematics equations for the palm, in conjunction with the equations produced by Wei et al. [3]. These are presented for reference in the beginning of this chapter.

A visualisation of the hand's workspace with a metamorphic palm is shown in Fig/ 6.6a. In contrast, Fig. 6.6b shows the workspace of the same finger structure but with a rigid palm. The fingers are attached on the exact same points on the palm and the palm geometry is the same. Just by examining the workspace of the index and thumb fingers and comparing it to the workspace of the same hand without an articulated palm, it becomes apparent how the metamorphic palm enhances the workspace of the hand compared to a fixed palm. The motion range of the fingers attached to the rocker link, the little, ring, and middle fingers, is still planar. However, it is enhanced by the rocker link of the palm.



(a) Workspace With Metamorphic Palm.

(b) Workspace Without Metamorphic Palm.

Figure 6.6: Workspace Generated Through Spherical Trigonometry.

The enhanced workspace of the thumb can be seen in Fig. 6.7. It is evident that the thumb, similar to the index finger, has a significantly enhanced workspace. The thumb workspace is no longer planar but is now spacial. This allows the thumb to have a great variety of configurations relative to the other fingers. This is particularly useful when grasping and when manipulating objects. This point is further reinforced in the origami folding experiment in chapter 12.

A 3D rendering of the hand is shown in Fig. 6.8. This is presented to the hand operator by the control software. Link geometry information, in the form of triangle vertices, is presented to the software by using a custom STL file parser. A C++ library containing the kinematic equations described in this dissertation is used to compute palm joint angles. These computed joint angles are then used to form homogeneous transform matrices based on the work of Wei [3]. These homogeneous transform matrices and vertices are then fed into a standard C++ / DirectX10 pipeline to render the hand on screen.

Figure 6.8a shows the dexterous operating mode of the hand where no palm joint is singular. In this motion branch, both palm input motors affect the pose of the hand independently and

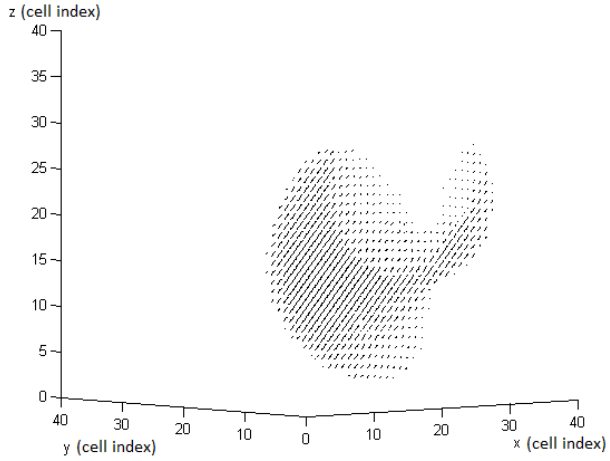
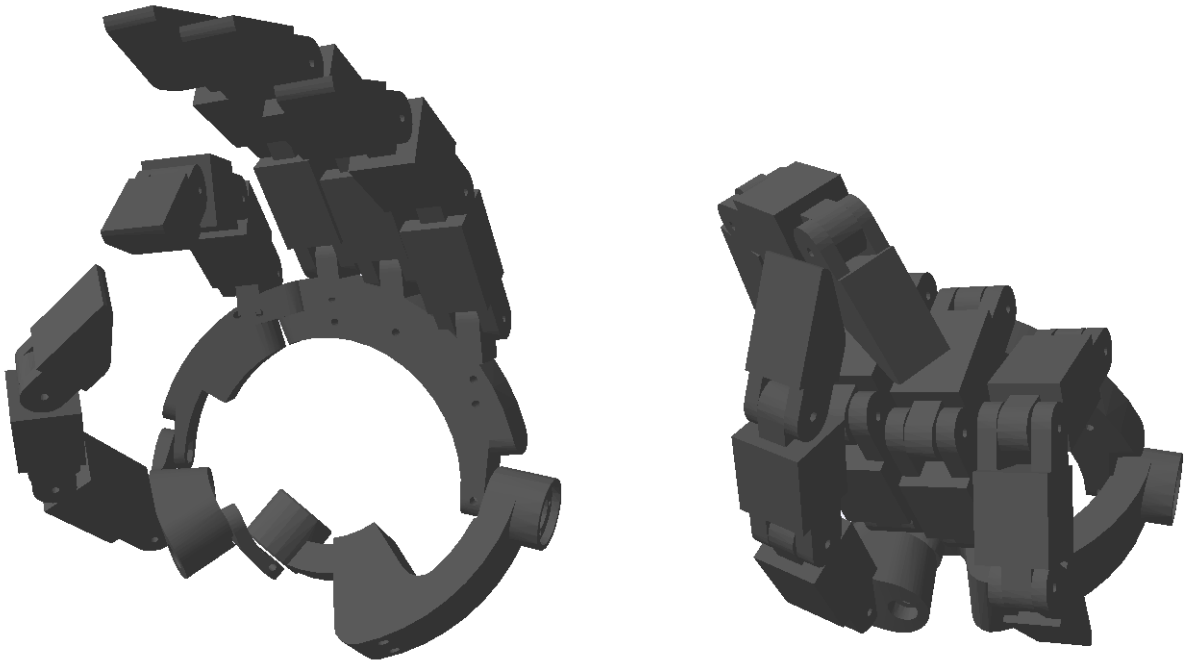


Figure 6.7: Workspace of the Thumb.



(a) 2-DOF Dexterous Configuration.

(b) 1-DOF Compact Configuration.

Figure 6.8: Hand Metamorphosis.

the hand shows maximum dexterity. This configuration is useful for manipulating large and complex objects, including articulated objects as shown in Fig. 11.10a. When the crank link α_1 folds over the ground link α_5, the palm becomes a 1 DOF mechanism. One such configuration is shown in Fig. 6.8b. In this configuration the hand becomes compact and can manipulate objects in tight spaces. It can also perform grasps that are more stable since less degrees of freedom need to be controlled. In such a 1-DOF configuration the fingers can easily collide and as a result the possible hand poses are limited.

6.7 Numerical Examples, Hand Poses, and Palm Configurations

To better illustrate the enhanced dexterity of a hand with an articulated palm, a number of numerical examples with the corresponding visualisation of the hand are given.

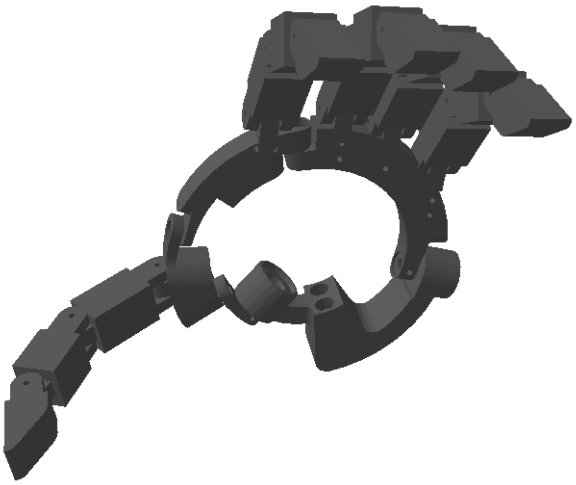


Figure 6.9: Hand with $\theta_1 = 0, \theta_5 = 0$.

The hand with the palm configured in a flat configuration with $\theta_1 = \theta_5 = 0$ is show in

Fig. 6.9. In this configuration the palm of the hand is completely open. It is primarily used when approaching or moving away from an object. Another use-case for this configuration is while manipulating large objects, such as large panels of articulated objects. While at this configuration, θ_1 can not change unless θ_5 changes first. This is because the combined length of the two coupler links is exactly the one needed to connect the crank to the rocker. If the crank rotates the crank-rocker distance increases while the opposite happens is the rocker moves.

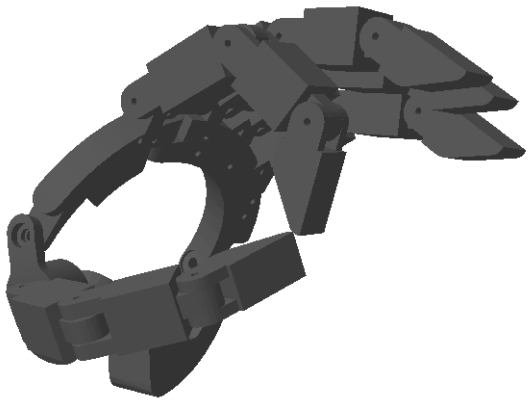
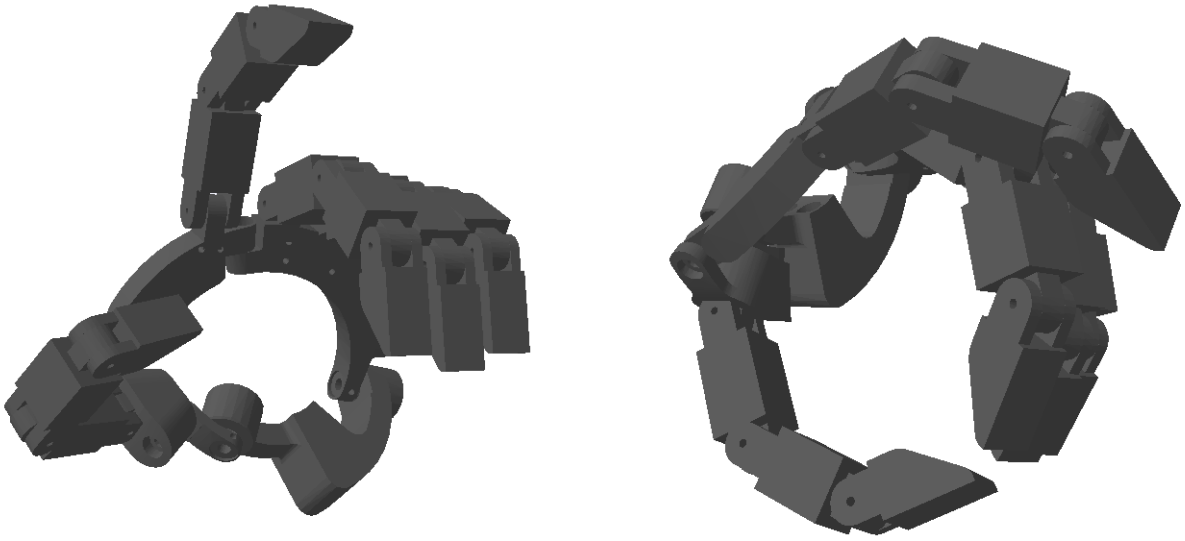


Figure 6.10: Pinch $\theta_1 = -34, \theta_5 = -32$.

Figure 6.10 shows the palm configured with $\theta_1 = 34deg$ and $\theta_5 = 32deg$. This is a configuration that looks more natural and very close to a relaxed human hand. In this case it allows the index finger and thumb to perform a pinch grasp. Further, at these angles there is enough separation between all the fingers to allow for dexterous manipulation of objects without much concern for collisions.

Two configurations of the hand primarily for grasping and performing in-hand manipulation tasks to cylindrical objects are shown in Fig. 6.11. In both cases a cylindrical cage grasp



(a) Cylindrical $\theta_1 = 0, \theta_5 = -60$.

(b) Cylindrical $\theta_1 = 180, \theta_5 = -90$.

Figure 6.11: Hand Metamorphosis.

is formed with the three grasping fingers, the middle, ring and little finger opposing another part of the hand.

In the first case, as shown in Fig. 6.11a it is $\theta_1 = 0deg$ and $\theta_5 = 60deg$. In this case the three grasping fingers and the palm wraps around the cylindrical object. The thumb is positioned in such a way as to potentially manipulate the side of the cylindrical object being grasped, depress the valve of a spray can for example. In this case the manipulation range of the thumb is limited. The index finger is kept open to avoid collisions with the middle finger as the palm changes to change the orientation of the finger operation plane of the thumb.

An alternative cylindrical grasp is whown in Fig. 6.11b. In this case the palm is configured with $\theta_1 = 180deg$ and $\theta_5 = 90deg$. The rocker link is positioned so the palm and fingers can partially wrap around a cylindrical, spherical or similar object. The coupler links are in such a way as to allow the thumb to oppose the grasping fingers. Further, just by only changing

joint angle θ_1 and rotating the crank, the thumb moves in such a way so as to oppose each one of the grasping fingers individually. This is done without moving the motors of the thumb. This is particularly useful when for example performing in-hand rotation of an object, such as a cup. The fingers maintain a grasp by pushing on the object. The orientation of the object is controlled by only actuating joint θ_1.

Table 6.1: Forward Kinematics Numerical Examples.

θ_5	θ_1	θ_{2nr}	θ_{3nr}	θ_{4nr}
		(deg)		
0	0	0	0	0
30	0	−1.22	30.59	0.87
60	0	−2.64	61.29	1.88
90	0	−4.61	92.25	3.28
120	0	−8.15	123.98	5.8
90	180	−169.15	62.12	58.13
120	180	153.1	107.05	62.55
−120	180	28.01	107.05	−148.87
−90	180	56.79	62.12	127.4

Numerical examples are shown in Tab. 6.1. Rocker input joint angle is limited to $\|\theta_5\|/leq120deg$. Crank input joint angle is first set to $\theta_1 = 0deg$ then to $\theta_1 = 180deg$ to first show the home configuration, then to show the 1-DOF compact configuration. The symmetry of the joint-coupler angle θ_3nr can be verified when rocker input joint angle θ_5 is changed to be reflex and non-reflex. Another feature particular to the non-reflex joint-coupler configuration is when $\theta_1 = 0$, the crank coupler joint angle θ_{2nr} and the rocker coupler joint angle θ_{4nr} change very slightly. Coupler joint angle θ_{3nr} is almost equal to the rocker input angle θ_5. This is because of the decision to be $\alpha_1 + \alpha_2 + \alpha_5 \approx \alpha_3 + \alpha_4 \approx \pi$ in this particular design.

Table 6.2: Inverse Kinematics Numerical Examples.

θ_{3nr}	θ_{4nr}	θ_5 (deg)	θ_1	θ_{2nr}
0	0	0	0	0
30.59	0.87	29.99...	0	-1.22
61.29	1.88	60.00	-0.01	-2.64
92.25	3.29	90.00	0	-4.61
123.98	5.80	120.00	-0.01	-8.15

Inverse kinematics numerical examples are shown in Tab. 6.2. The joint-coupler angle θ_3 and the rocker coupler joint angle θ_4 are given. For ease of comparison, the values are drawn from the first five rows of Tab. 6.1. Then, as described in Sec. 4.7, a cyclic permutation of the dimensions of every link is performed. The problem is solved with the value of the permuted θ_1 corresponds to θ_4 of Tab. 6.1. The value for the permuted θ_5 corresponds to θ_3 of Tab. 6.1. After the computations, the mechanism joint angles are remapped following the inverse cyclic permutation. The results are presented in Tab. 6.2. The small numerical deviations are due to the rounding of the values of the permuted input joints to the second digit. This cross-validation procedure was also used during the development of the method to insure correctness.

6.8 Conclusions

To better understand the way a metamorphic palm can improve the workspace and dexterity of the hand, first a method for generating the joint space of the palm and the workspace of the hand are needed. Existing works did not differentiate the joint-coupler reflex and non-reflex

configurations. These methods also fail in the case of a singularity, even though the mechanism is still controllable.

Spherical trigonometry-constrained kinematics and the previous work of Wei et al. are used to produce the joint space and work space of the hand, along with a workspace visualisation of the palm and hand. Point clouds are generated that represent the joint space of the palm and the workspace of the hand. The joint space as well as the workspace point clouds were generated and stored in a computer using an efficient method of segmenting the workspace into 32 bit word cells. Each bit of each cell encodes a different point of interest of the hand. A comparison of a hand with a metamorphic palm and the same hand with a fixed palm was made. A number of examples of the two DOF dexterous configuration and the one DOF compact configuration were shown.

Plotting the joint space of the palm revealed the usefulness of the joint-coupler reflex and non-reflex configurations. It was shown that selecting one configuration or the other allowed significant change in the palm configuration while keeping the angle between the index finger and middle finger the same for a large range of motion. This is useful for example when grasping cylindrical objects and manipulating part of the object with the thumb. It was also shown that the joint-coupler singularity planes in the joint space act as transition from the reflex to the non-reflex configurations.

By using spherical trigonometry-constrained kinematics it is now easy to differentiate the joint-coupler reflex and non-reflex configurations. This in turn, along with the simple nature of the math involved, allows for easy simulation of the palm and hand in the joint space and workspace. Joint space and workspace simulation is key in understanding the capabilities of the metamorphic hand in terms of dexterous object manipulation. The contrast between the workspace example of a hand with a fixed palm and the same hand with a metamorphic palm

highlights the advantage of an articulated palm equipped metamorphic hand.

Chapter 7

Experimental Set-Up

7.1 Introduction

Following-up on the new developments on the kinematics, joint space, and workspace of the metamorphic palm, a new robotic hand was designed. The goal was to illustrate the capabilities of robotic hands incorporating the metamorphic palm in their design. A physical prototype of a new metamorphic robotic hand, capable of dexterous manipulation of articulated objects, was developed.

Developing the sensing and control system for a robotic hand is no simple task. A robotic hand consist of many sub-systems, namely the digits, the sensing, the actuation, and the control system. Further to normal robotic hands, a metamorphic hand also has an articulated palm. It is hard to develop and tune a control system when this many inputs and outputs, all operating concurrently.

In order to develop the physical prototype for a new metamorphic hand, it was necessary to first study earlier prototypes. Practical issues were identified and solutions were assessed.

Then it was appropriate to develop experimentation apparatus to aid the investigation of different sensing and actuation systems. It was important to focus on single tendons and joints in isolation, before continuing development of a fully integrated robotic hand.

This chapter starts by showing the failure modes of the tendon drive system of early prototypes. It then progresses by presenting the experimentation platform used in the development of the sensing and control system.

7.2 Problems in the Tendon Drive System of the Metamorphic Hand

The new hand had to be more reliable and tolerant to manufacturing and assembly defects.

Figure 7.1: Broken Tendon.

Figure 7.1, of a previous version of the tendon drive system of the KCL metamorphic hand [3, 37, 74, 75], shows a broken tendon. This was the main failure mode of the hand actuation

system. When the tension of the tendon increases past a certain threshold, identified by the tendon manufacturer, the tendon will break.

One occasion where a tendon can break is when the finger joint the tendon is acting upon has reached a mechanical motion limit. If the actuator moves in order to increase the tension on the tendon, the tendon sheath will contract and the tendon will break.

A second case is when the actuated digit tries to move an object that requires a force larger than the tendon can bear. In this case, the PID controller controlling the digit will steadily increase the applied force, thanks to the integral part of the controller. When the force is too large, the tendon will break.

A third case is when the digit is not actuated but an external disturbance is injected into the system, maybe because another object collides with the finger. If this disturbance is large enough, it may generate a tension adequate to break the tendon.

In all these circumstances, the control system needs tendon tension feedback. It also needs to be able to reject external disturbances well enough to prevent tendons from breaking.

The second failure mode identified was loose tendons. An example of a loose tendon is shown in Fig. 7.2. The second tendon from the left is no longer under tension.

One of the reasons which can lead to a loose tendon is an incorrect amount of pre-tension during assembly. As a result, the pre-tension mechanism does not have enough travel to keep the tendon under tension and the tendon comes loose.

Another reason for a tendon to lose tension is the wearing out of the tendon pulley to motor shaft interface. If the pulley is attached to the shaft through a locking screw then this does not happen. This however requires fixed positions for the screw to lock into. One of the two tendon pulleys has a locking screw and so this particular tendon never comes loose. The other pulley however needs to be secured in place by means of friction to enable proper

Figure 7.2: Loose Tendon.

tendon tensioning. If after tensioning the second pulley is not secured in place, it can slide. This sliding of the pulley allows the tendon to become relaxed.

The third reason is improper fitting or wear of the tendon sheaths. In this case all pulleys are well locked into place. The tendons are properly pre-tensioned. However, the tendon sheath slides deeper into it's slot after assembly. This has the effect of shortening the tendon path and thus allowing the tendon to lose tension.

Figure 7.3, again of the KCL metamorphic hand, shows a close-up view of an extreme case of a tendon-pulley motor-shaft interface failing. As the motor is applying torque to the faulty pulley, the pulley slides on the motor shaft. With no tendon tension sensing and only joint angle sensing, the motor will keep rotating in order to move the joint. As the motor continuously rotates to tighten the tendon attached to the faulty pulley, it keeps unwinding

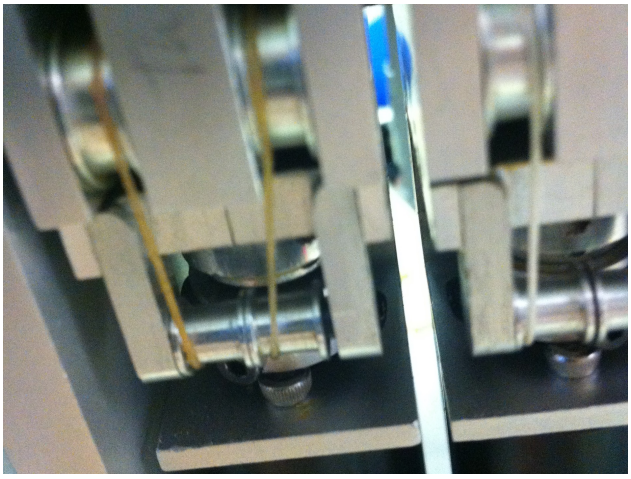


Figure 7.3: Loose Pulley.

the other tendon on the good pulley. If this keeps happening, then the good tendon will unwind completely, and then will start winding in the opposite direction. The result is both tendons end up being wound up in the same manner. The tendon attached to the good pulley will eventually brake if the motor keeps rotating in the same direction. Alternatively, they will both unwind if the motor then rotates in the opposite direction. In both cases, the result will range from catastrophic to the need to disassemble and re-assemble the hand.

7.3 Tendon Layout in the Hand

Tendons are a good force transmission alternative to linkages, shafts, gears, pneumatics, and hydraulics. Geared mechanisms introduce a lot of friction and backlash in the system. Shafts are rigid and require a complex combination of joints to transmit torques when the mechanism is articulated. Linkages have non-linear force transmission profiles and take up a lot of space compared to the other options. Pneumatics require the presence of an air compressor and hydraulics can leak. Both pneumatics and hydraulics need sizeable and complex valves and

pistons that are hard to fit inside of the fingers of a dexterous robotic hand. Hence tendons offer the best alternative when space is limited and there is a need for flexibility, as in the case of the metamorphic hand with the articulated palm.

When installed on the real hand, the number of introduced non-linearities significantly degraded the performance of the single axis control system. One reason for this degradation in accuracy is the need for the tendon tensions to vary significantly and as such the two tendons are working outside of the linear region. Further, the tendon and sheath stiffness curve relies on environmental conditions, the shape of the tendon path, as well as tendon and sheath wear. Moreover, the tendon sheaths are free standing. As such, their parameters change randomly, as the posture of the palm changes. This reduces the practicality of using a joint position estimator as the one described for small force variations. A state observer with a full tendon and transmission model based on the dynamic behaviour of the transmission system is also not feasible since the parameters of the system change randomly based on how the tendon sheaths interact with the environment.

This led to a decision to revise the tendon layout and motor assembly design. The two alternatives are the tendon layouts usually referred to as “2N” and “N+1”. Both have advantages and disadvantages. The “2N” tendon layout is able to provide the highest degree of control for joint torques. Since the torque on each joint is dictated by the difference in tendon tension and each tendon is independently controlled. This design however requires the highest number of motors. Equal, as the name suggests, to twice the number of joints. This not only raises the weight of the hand, it also complicates the motor communication system and increases sampling time and cost since more motors need to be controlled. The “N+1” tendon layout is a good alternative. It uses the least number of motor-tendon pairs, only one pair more to the number of joints. The disadvantage of that scheme however is significant

coupling of forces among joints, as well as larger nullspace forces.

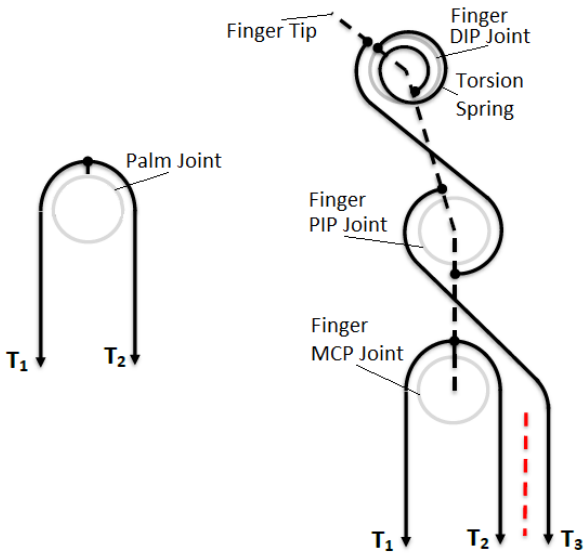


Figure 7.4: Tendon Layout.

A hybrid tendon layout was used, as shown in Fig. 7.4. It is based on the work of Palli et al. on the UB hand [71]. Palli et al. used a "N+1" scheme with the PIP and DIP joints coupled using a passive tendon. Here, the importance of each joint was assessed based on its place in the joint hierarchy of the hand. Moreover, the digits were grouped into a dexterous set and a grasping set. The most important joints, dictated by the joint hierarchy, would be actuated using a "2N" tendon layout. This includes the palm joints as well as the CMC and MCP joints of the digits. The PIP and DIP joints would be actuated by a single tendon, antagonised by a spring. The fact tendon T_3 is not coupled to the MCP joint is highlighted in Fig.7.4 by the red, dashed line. Since this would be an under-actuated system, a passive tendon was added. This tendon strongly couples the PIP and DIP joints of the dexterous digits. Finally, the grasping digit is controlled by a single tendon and spring system.

7.4 Tendon Drive System Experimentation Set-Up

In order to facilitate further developments and experiments on the control system of the hand, new experimentation devices needed to be designed and manufactured.

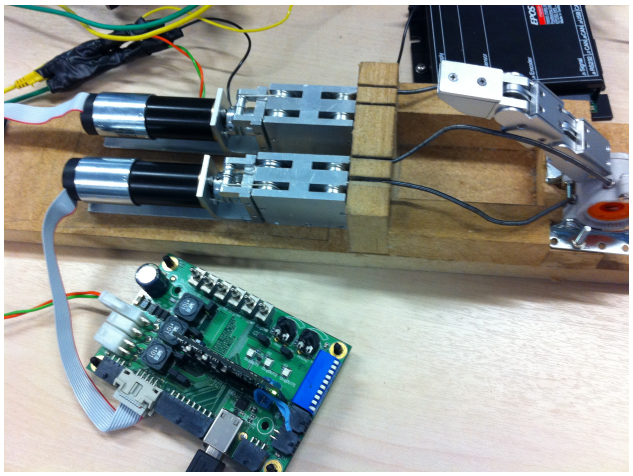


Figure 7.5: Actuation Assembly Experimentation Rig.

Figure Fig. 7.5 shows the experimentation apparatus designed and built for the purpose of further developing the sensing and actuation of the hand. The material chosen was MDF since it was readily available and easy to work with. Measurements were taken of the ECHORD DEXDEB actuation assemblies. Based on these measurements, the size of the fingers, the length of the tendons, and the length of the tendon sheaths, the size of the base of the experimental rig was decided. Blocks were then placed, glued, and secured in place with screws at the appropriate locations. The aim was to enable easy access to all elements of the finger actuation system. Appropriate mounting holes were drilled so each component could be held in place securely. The interchangeability of different components facilitated further experiments.

The first section of the rig is designed to hold up to two motor and sensor assemblies.

This is to enable the development of an antagonistic system. The second section is designed so it can act as an anchor point for tendons sheaths of different sizes. The third section is designed so a finger and other mechanical components can be attached with the aim of further developing the control system of the hand.

7.5 Conclusions

This chapter presented problems identified in earlier physical prototypes of metamorphic hands. A new experimental platform was built to study tendon and joint sensing and actuation in isolation. A the new tendon layout was developed and used in the new tendon driven control system for the metamorphic hand.

The tendon layout is designed for strong coupling of the intermediate and distal interphalangeal joints in the fingers, while still being compliant to prevent tendon damage. The experimental set-up is designed to allow mounting of the key components of the tendon drive system, while allowing easy access to the components for easy experimentation. This enabled gathering of repeatable and useful data.

By studying the previous versions of the metamorphic hand, the most common failure modes of the tendon drive system were identified. It became apparent there is a need to sense tendon tensions, even in an open loop tendon drive system. This sensing enables the control system to prevent tendons from breaking and to avoid turning the motors when the tendons are loose. Failure to do so results in tendons becoming entangled and cause damage to the tendon drive system components. It was also found it was better to use two different motor assemblies and control each tendon individually.

The experimental set-up was created for the purpose of developing a new control system. It enabled quick and easy assessment of a number of different sensing and control schemes,

both for tendon tension and joint position. The experimental jig allowed data collection in a repeatable manner by reducing the number of system parameters. It also allowed work to be focused on a single tendon or joint at a time.

Chapter 8

Tendons and Sensing

8.1 Introduction

As discussed in chapter 7, early prototypes of the KCL metamorphic hand used an open loop control system. That system was relying on estimating the position of each finger and palm joint based on the positions of the actuators. Such a system would often experience numerous faults in the tendon drive system and the performance would be limited. Further, this system did not allow for any kind of force control. Handling of delicate materials was not always possible. Further, there were a number of faults directly attributed to the tendon tension not being sensed. This indicated the need to add tendon tension sensing capabilities to the hand.

Tendons are liable to breaking or even coming loose. They have to be constantly monitored to avoid failures. If the tendon tension becomes too high, a tendon will break. If tendon tension becomes too small, the tendon will very likely come off a pulley, slack, and even become entangled. The consequences of these events can be catastrophic. Measuring the tension on a tendon also allows for more advanced control schemes, such as force control. Being able

to measure the tension on the tendon enables the control of this tension. Hence, sensing the tension of a tendon is very important.

Sensing the tension of a tendon can be accomplished in many ways. It is usually measured indirectly, by the effects it has on some other measurable quantity. The most typical method is to have a tendon pulley on a bendable beam. Then the strain of the beam is measured by a strain gauge. The tendon tension is then inferred based on the stiffness of the beam, the point of contact to the tendon and the measured strain. These methods however are limited in the sense that once a the beam material is chosen and the stiffness of the beam is designed based on the anticipated tendon tensions, the sensing element has very specific range and sensitivity.

The sensor presented in this chapter overcomes the need for a beam. Instead, it uses a transducer with high dynamic range. This allows measurements of both small and large tensions, with a loss in resolution in the high tension range. This kind of sensor is ideal in the case of the hand. It allows bias forces to be small due to its high resolution in the lower ranges but can also measure large forces if required. Further, a process is shown for tendon stretch estimation.

8.2 Tendon Tension Sensor Design for the Metamorphic Hand

A preliminary experiment was performed to assess the feasibility of a design candidate. One of the previous tendon actuation assemblies was used. A pressure sensitive resistor was used, along with a small metal block as a tendon-transducer interface.

The force sensitive resistor (FSR) that was used as the main sensing element for measuring tendon tension is shown in Fig. 8.1. The resistance value of the FSR is reduced when pressure

Figure 8.1: Pressure Sensitive Resistor.

is applied on its disk. The resistance value is inferred by connecting the resistor in a typical voltage divider circuit. Then the output voltage of the circuit is measured. The FSR consist of three layers. The bottom layer is a flexible substrate with printed semiconductor material. This layer provides the resistance to be measured. The second layer is a spacer, separating the top and bottom layers. Adhesive is applied on both sides of the second layer. The second layer provides an opening through which the top and bottom layers make contact, depending on the applied pressure. There is also a second cut along the tail of the second layer. That cut forms a vent to allow air to flow freely when the sensor is compressed. Finally, the top layer consist of a flexible substrate and a printed pattern of a conductive material. The print is in such a way so the resistance of the sensor changes when pressure is applied.

Tests were performed to assess the suitability of the FSR as a tendon tension sensing element. The experimental apparatus described in chapter 7 was used, as shown in Fig. 8.2. The actuation assemblies from the ECHORD DEXDEB version of the KCL Metamorphic Hand were mounted on the test rig. A pulley is added which rests on the FSR and bears the

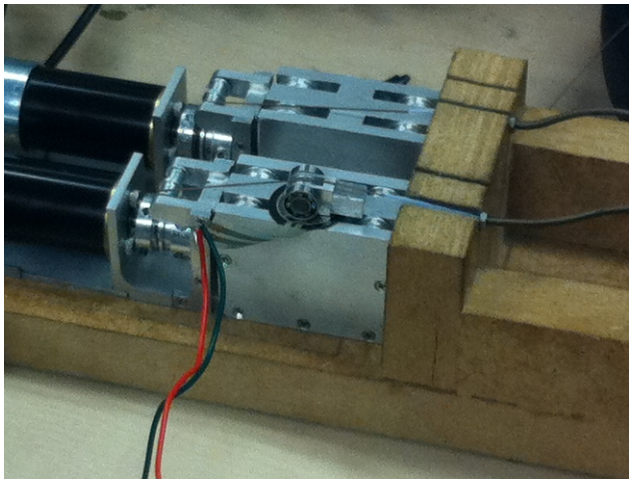


Figure 8.2: Preliminary Tendon Tension Sensor.

tendon. As tension on the tendon is increased, the force from the pulley to the FSR increases. This force then changes the resistance of the FSR. Direct contact of the tendon with the sensor would damage the sensing element. A metal block is used to evenly distribute the force on the the sensor. The tests showed the resistance of the sensor varied adequately with a change in the applied force.

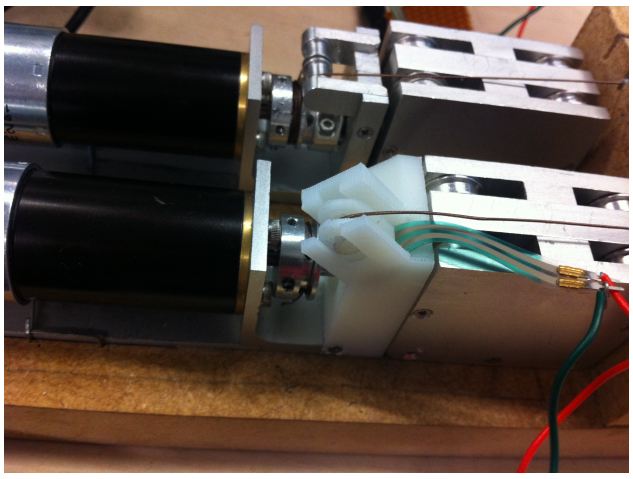


Figure 8.3: Preliminary Sensor Assembly Design.

Following the results of the preliminary tests, a new sensor mount was designed, as shown in Fig. 8.3. The component was designed so it could be inserted in the existing tendon actuation assemblies. This was accomplished by having the depth of the device span the distance from the tendon pulley to the tendon pre-tension assembly. A special slot was designed to hold the sensor. Care was taken to not bend the tail of the sensor more than the value specified by the manufacturer. A tendon-sensor interface was designed in the form of a block. The sensor side of the block is circular and with rounded edges. This ensures good contact with the sensor but without damaging it. The tendon side has a pulley onto which the tendon slides. This way, friction is reduced to not affect the measurements significantly.

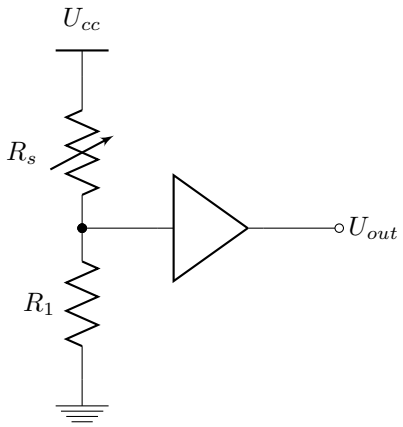


Figure 8.4: Sensor Circuit.

The schematic diagram of the circuit driving the sensor is shown in Fig. 8.4. The sensing element is the force sensitive resistor. It is connected in series with the resistor R_1. Together, they form a voltage divider. An operational amplifier is configured as a voltage buffer. It is used as an interfacing component to an analogue to digital converter (ADC) peripheral of a micro-controller (μC). The micro-controller in turn runs special software designed to sample sensor data from the ADC. The data is then transmitted to the rest of the control

system through a serial interface over a universal serial bus (USB) cable. The details of the communication protocols and algorithms involved are discussed in chapter 11.

8.3 Tendon Stretch Based Joint Position Estimation

Calculating an exact value in Newtons for the tension of a tendon at the actuator end is not meaningful. This is due to the tendon sheath exhibiting an almost chaotic behaviour. The value measured at the actuation end of the tendon does not correspond well to the tension at the joint end of the tendon.

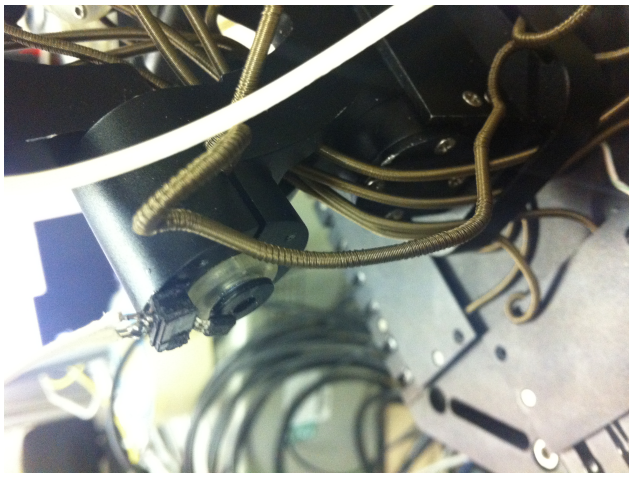


Figure 8.5: Tendon Sheath Deformation.

The tendon sheath deforms as the tendon tension is increased. One such deformed sheath is shown in Fig. 8.5. This deformation renders friction and stiffness of the sheath non-linear. As such, calibrating the sensor to indicate force in Newtons is impractical for this application. It was observed that tendon stretch had a strong correlation to the measured tension. Further, the tension-stretch profile is constant during the course of a single operation. It was thus deemed beneficial to calibrate each sensor for tendon stretch. A process was also developed

that would yield repeatable measurements, so it could be performed each time the tendon sheath characteristics would change.

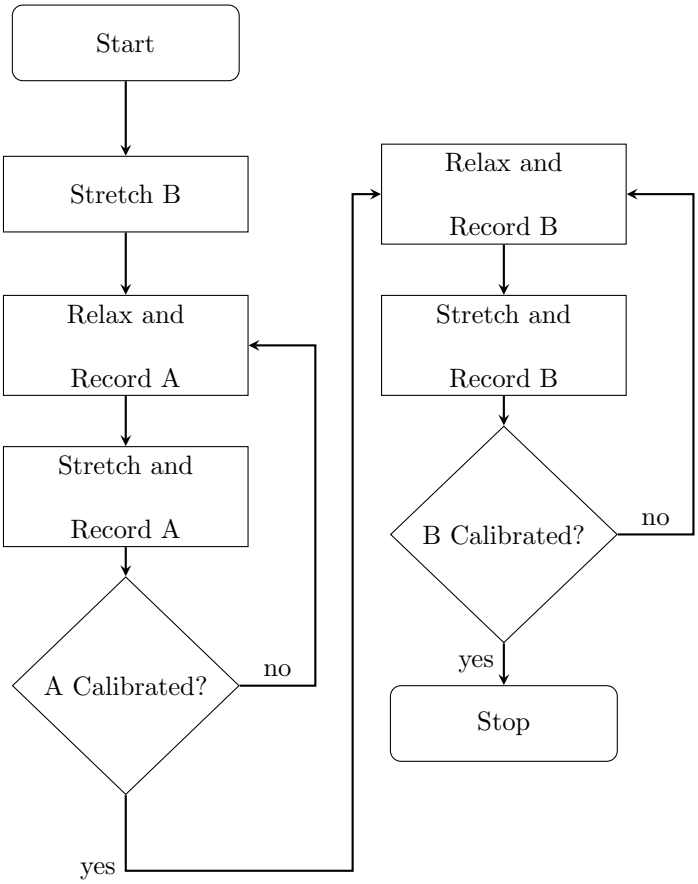


Figure 8.6: Tendon Stretch-Tension Calibration Algorithm.

The tendon stretch calibration algorithm is shown in Fig. 8.6. This algorithm is designed for antagonistic joints. First, one tendon tension is set to a minimum safety value. The other tendon force is increased to a maximum value. The system is actuated in this manner until all motion stops. This indicates the joint is at a mechanical limit. Then the tendon with the smallest force is stretched. The displacement sensor output curve is recorded. The process is repeated in the other direction and for both tendons.

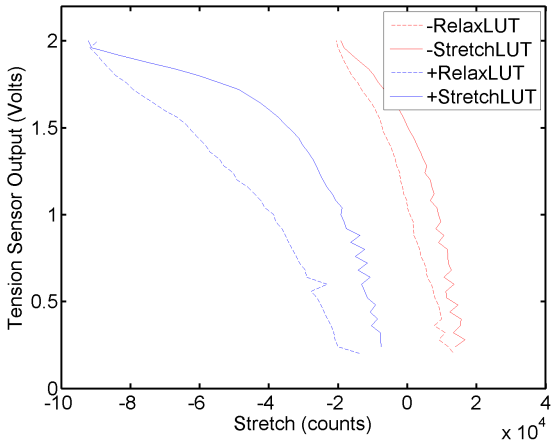


Figure 8.7: Tendon Stretch Calibration Curve.

Figure 8.7 shows the data obtained from the calibration process for one tendon. In order to linearise the sensor output, this data is stored in a look up table. It is then used to interpret the voltage measurements from the sensor into a value representing the stretch of the tendon.

An estimation of the joint position is performed based on the transmission model. The actual position of the joint was recorded by using a protractor as seen in Fig. Fig. 8.8.

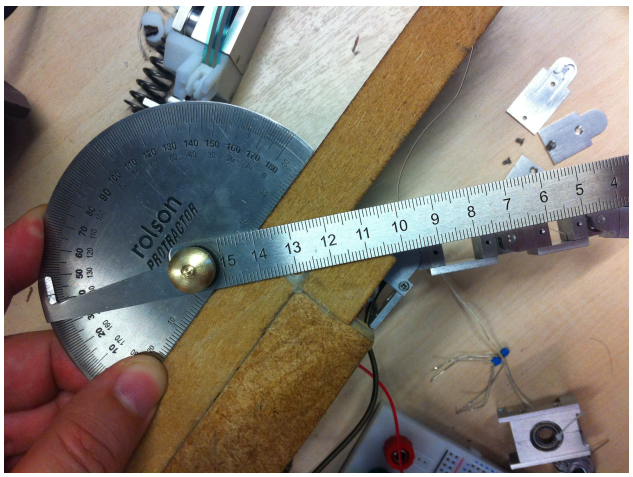


Figure 8.8: Measuring Joint Position.

Table 8.1: Joint Angle Estimation for Measured $\theta = 48deg$.

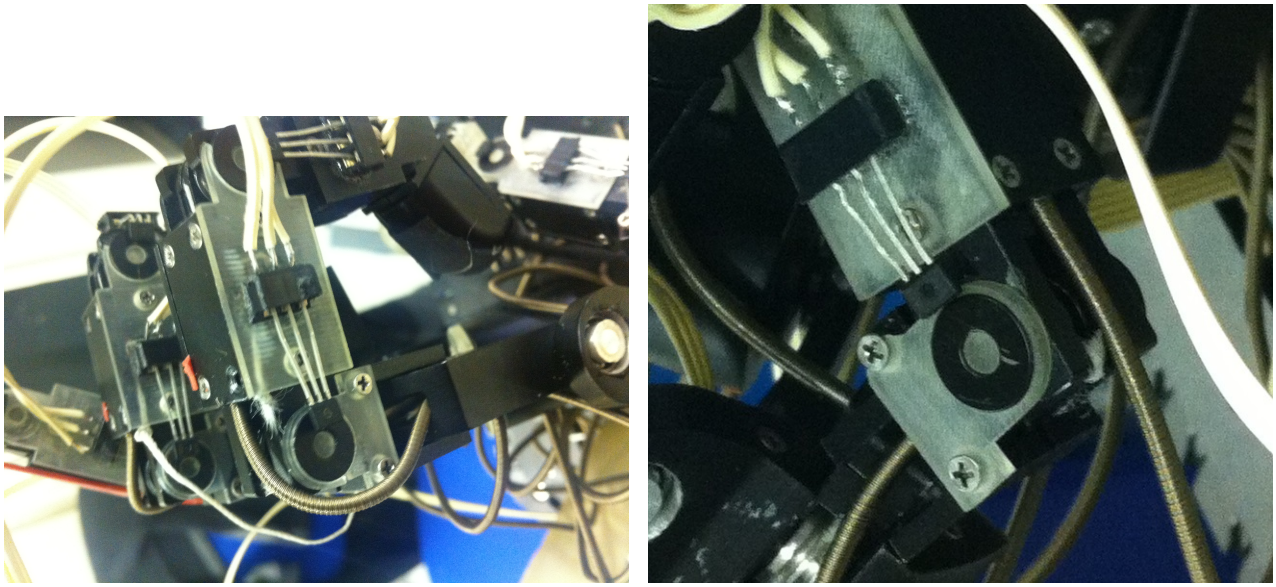
V_{tot}	a	V_1	V_2	x_1	x_2	$\hat{x}$	$error$
		$(Volts)$		deg		deg	$\%FS$
4.2	0.5	2.1	2.1	44	52	48	0.00%
4.2	0.45	1.89	2.31	43	51	47	1.11%
4.2	0.4	1.68	2.52 45	50	48	49	0.00%
4.2	0.35	1.47	2.73	50	48	49	1.11%
4.2	0.3	1.26	2.94	60	49	55	7.77%

A number of measurements were taken. Table Tab. 8.1 shows a comparison of the estimated joint position compared to the measured joint position. It can be seen that when tendon forces are varied in a small region, the linear approximation is valid and the joint position is correctly estimated. When the force variation is large, the estimation error becomes large and the result of the estimation impractical.

8.4 Hall Effect Sensors for Joint Position Sensing

An alternative is to use hall effect sensors. Hall effect sensors are based on the hall effect. The voltage across the terminals of the sensor is proportional to the current flowing trough the sensor and the strength of the magnetic field perpendicular to the sensor.

Figure 8.9a shows a poorly installed hall effect sensor while Fig. 8.9b shows a hall effect sensor installed properly. In the first case the change in magnetic field is small compared to the change in joint angular displacement. In the latter case, the change in magnetic field strength is much larger for the same change in angular displacement. This results in better signal to noise ratio and an improvement in the quality of the measurements. One advan-



(a) Hall Effect Sensor, Bad Placement.

(b) Hall Effect Sensor, Good Placement.

Figure 8.9: Hall Effect Sensors.

tage of hall effect sensors over potentiometers is the magnet and the sensor do not share a mechanical connection. This eliminates stress due to joint axis misalignment and provides high tolerance during assembly. Since the output of the Hall effect sensors is not linear, a number of approximately linear regions are chosen and the sensors are calibrated using a set of calibration jigs. This process generates five measurements at five different positions. These are then used when interpolating the sensor reading to estimate joint position.

8.5 Conclusions

This chapter presented the tendon tension and joint position sensing sub-system of the hand. Tendon tension sensing is not only important for providing feedback in a control loop, but to also safeguard against over-tension and under-tension. These events can have catastrophic

consequences, as shown in chapter 7.

Tendon tension is measured by means of a pressure sensitive resistor, integrated into the tendon actuation assemblies. A method was presented to estimate joint position based on the tendon tension. Finally, hall effect sensors were used for more accurate joint position sensing for the joints of the palm and fingers of the metamorphic hand.

The transmission system is highly non-linear and exhibits an almost chaotic behaviour. There is limited use for calibrating the sensors for sensing the exact tension on the tendons. The tension on the joints of the fingers is greatly varying due to friction inside the tendon sheaths. Instead, the measurement from the tension sensors is used as a limit value to prevent breaking, for estimating tendon stretch, as a rough estimation of joint torque, and as a means of closing the control feedback loops.

By using force sensitive resistors there is no need for a cantilever beam and strain gauge-based tendon tension sensor. This allows tight integration of the sensor to the motor assembly. This also offers a higher dynamic range and no limitation to the used material and geometric properties of the tendon bearing and tendon-sensor interface. The disadvantage is the measurement is not linear. As such time consuming calibration of every sensing element is necessary, if accurate measurements are required.

Chapter 9

Metamorphic Hand Actuation

9.1 Introduction

The metamorphic palm mechanism is intended to be used as the basis for metamorphic hands. The palm is not a fixed mechanism. It is articulated. And so, it is hard to fit gear mechanisms and to use rigid axes for transmission of force and motion. Direct drive for the fingers is not an option for this particular prototype. It designed for origami carton folding, as is shown in chapter 12. It had to be human sized. There are currently no motors available small enough and with sufficient torque output, appropriate for the task.

The most practical method to transfer forces from the base of the hand to the hand digits, is through a tendon drive system. Chapter 8 described the sensing scheme for the hand. This chapter focuses on the design of the control system. The limitations of a typical PI control system are explored, then two improved systems are presented. One hysteresis control system with compensation for tendon tension is introduced. An alternative PI control-based system with friction and tendon tension compensation, but more complex to tune, is then presented.

Finally, a multi-layer design for systems with many digits is presented.

9.2 PI Control Limitations in the KCL Metamorphic Hand

To better understand the limitations of PI control in the context of the KCL metamorphic hand, picture the following situation. Suppose there is a one dimensional space represented by a line. A moveable point on that line represents the controlled variable. Another fixed point represents the desired value of the controlled variable. A P controller would be a mechanical spring with each of its ends attached to each of these two points. The further away the controlled variable is from the desired set point, the higher the action of the spring. If the system were to be set in motion, the controlled variable would oscillate about the set point indefinitely. If viscous friction were to be introduced into the system, the controlled variable would eventually settle at the set point.

Now, another spring is introduced. One end attached to the controlled variable and the other end attached to another point on the line. Friction is still present. At steady state, when all motion has stopped, the controlled variable will be at an equilibrium position, as by 9.1

$$k_1 \left(x_{CV} - x_{SP}\right) = k_2 \left(x_{CV} - x_2\right) \tag{9.1}$$

where x_{CV} and x_2 are the coordinates of the controlled variable and the second attachment point of the second spring. x_{SP} are the coordinates of the set point.

In order for the difference $x_{CV} - x_{SP}$, also known as the tracking error, to be reduced to zero, a term is necessary to negate the action of the second spring. In traditional PI control, this is accomplished by having a term which integrates the error over time. The controlled variable eventually reaches the desired value. In this case, the steady state would be as shown

in Eqs. 9.2, 9.3

$$k_1 (x_{CV} - x_{SP}) + I_{SS} - k_2 (x_{CV} - x_2) = 0 \tag{9.2}$$

and so

$$I_{SS} = k_2 (x_{SP} - x_2) \tag{9.3}$$

This type of control is well studied. If changes to the set point are small, this type of control behaves well. If however the set point changes significantly, PI control effectiveness is weakened.

For the case of the tendon driven system of the KCL metamorphic hand, when the controlled variable is at the set point, the proportional action is zero. The controller relies on the integral action to maintain the set point. When a new set point is set, the integral has to unwind for the controlled variable to reach the new set point. Since the K_I gain is kept small for stability, this process is slow and performance is poor.

9.3 Bang-Bang Control with Schedule to Neutralise Tendon Tensions

Better results were observed by using a bang-bang controller with step scheduling and a deadzone. A set of values corresponding to controller action is generated through trial and error. These are adequate to drive the tendon motors and tension the tendons to the desired set point with a desired transient response. The motors the tendons are attached to are thus rotated until the tendon tension is at the desired value. The maximum desired tracking error is chosen and defines a deadzone; a range of values about the set point where motor action

can stop.

$$C(e) = \begin{cases} 0 & \text{if } SP - DZ < e < SP + DZ \\ C_{BB}(x, \dot{x}) & \text{if } e \leq SP - DZ \text{or } e \geq SP + DZ \end{cases} \tag{9.4}$$

where $C_{BB}(x)$ is according to the schedule

$$C_{BB}(x) = \begin{cases} c_i & \text{if } x_{i-1} \leq x < x_i, \dot{x} \leq 0 \\ c_i' & \text{if } x_{i-1} \leq x < x_i, \dot{x} > 0 \end{cases} \tag{9.5}$$

The schedule values are selected such that they are adequate to overcome tendon tension and friction while at the upper bound of each schedule entry. The schedule steps are selected so no oscillations appear at the lowest bound of each step. This is done through trial and error. First starting at the highest tension setting. Then decreasing the set point until oscillations occur. Then increasing the set point until oscillations stop, plus a safety margin of 20%. The process is repeated until all schedule entries are filled and there are no oscillations while at the smallest tension set point. Two different sets of values are needed, one set for stretching and one set for relaxing the tendon.

The tension tracking performance of the bang-bang with scheduling and deadzone control scheme is shown in Fig. 9.1. The controller has a good step response with no overshoot and no oscillations.

In the first part of the figure, before the first step command is issued, there is a low frequency sawtooth waveform. At first, the tendon pulley is rotated by the motor until the tendon tension is within the deadzone that surrounds the set point. The motor is then no longer powered. The tendon tension is kept at that value by the friction in the motor gearbox. If tendon tension is strong enough to overcome gearbox friction, the pulley will start to slide. Once the tension is outside the deadzone, the motor is actuated until the tension is near the

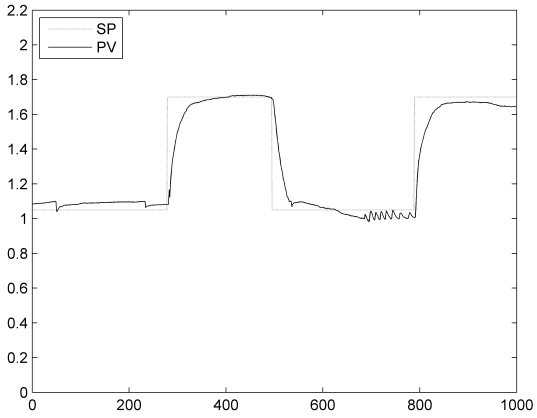


Figure 9.1: Tension Tracking.

desired value. This cycle repeats. It is desirable to prevent these oscillations. The control action value must be selected so within one sampling period, the motor movement is not large enough to change the tendon tension from one end of the deadzone to the other end.

At about sample number 700 and up to sample number 800, a disturbance in injected into the system. This is accomplished by moving the attachment point of the tendon to the finger. The controller behaves as desired and keeps the tension of the tendon within the deadzone.

9.4 PI Control with Feed-Forward Friction and Tendon Stiffness Compensation Schedule

Very good results were obtained by using a combination of bang-bang control and PI control. A gain schedule was used to negate the spring force at different operating points. The signal was fed forward and added to a PID controller. The PID controller corrects the mismatches due to friction and stiffness variation because of non-linearities, environmental changes, and

wear.

$$C(e) = \begin{cases} 0 & \text{if } SP - DZ < e < SP + DZ \\ C_{FF}(x) + K_P e(t) + K_I \int_0^t \mathrm{e}(\tau)\mathrm{d}\tau & \text{if } e \leq SP - DZ \text{or } e \geq SP + DZ \end{cases} \tag{9.6}$$

where $C_{FF}(x)$ is according to the schedule

$$C_{FF}(x) = \begin{cases} c_1 & \text{if } x < x_1 \\ c_2 & \text{if } x_1 \leq x < x_2 \\ c_3 & \text{if } x_2 \leq x < x_3 \\ c_4 & \text{if } x_3 \leq x < x_4 \\ c_5 & \text{if } x_4 \leq x \end{cases} \tag{9.7}$$

The schedule values are such that the applied torque of each entry is slightly lower than the sum of all ther forces on the motor. These include the force applied by the stretched tendon on the pulley and the gearbox and motor friction at the bottom end of each schedule window. This way the feed forward term minimises the effort of the integral part of the PID controller. This improves performance by reducing the damping coefficient of the physical system. Tuning of the feed forward path is done through trial and error. Tuning of the PI controller is done by combining the Ziegler-Nichols method [87] with trial and error.

The tension tracking capabilities of the system are highlighted in Fig. Fig. 9.2. The sharp peaks at the first, second and last change in set point are due to the disturbance caused by actuating both tendons of the joint. Compared to Fig. 9.1, the disturbance is rejected almost immediately and no oscillations occur. The addition of PI control allows the deadzone to be smaller. This enables more accurate tracking. The PI controller automatically changes the control effort to the correct value. This is an improvement compared to having a constant

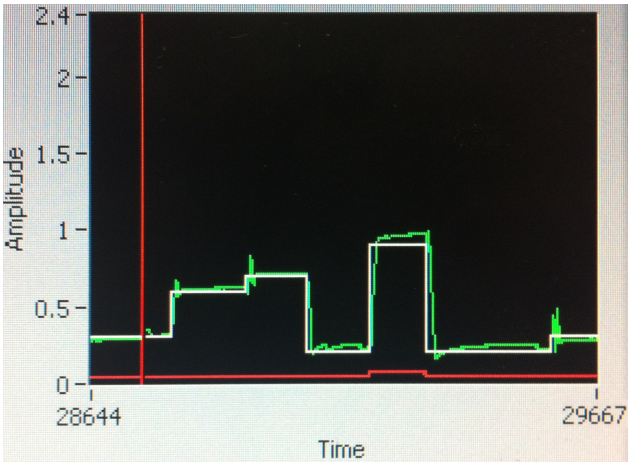


Figure 9.2: Tension Tracking performance.

control effort for a range of set points. The disadvantage of this method is a more involved tuning process than with bang-bang control.

9.5 Multi-Layer Control System

A multilayer control system architecture was chosen. Each layer of the control system is designed to linearise the part of the system it controls. This kind of modular controller design allows for easy modification and expansion of the control system as requirements change. Every layer of the controller is agnostic to the rest of the system. All that matters is the interface between layers. An ideal layer would present the system to the top layer as a linear system.

The block diagram of the control system of the KCL metamorphic hand is shown in Fig. 9.3.

Motor controllers are at the lowest level. Inputs to this block is in the form of motor position, velocity, and current. These are PID based control boards provided by the motor manufacturer. They are capable to control motor current, velocity, or position. If set to current control mode, finer control of the tendon force is possible. However, as number of

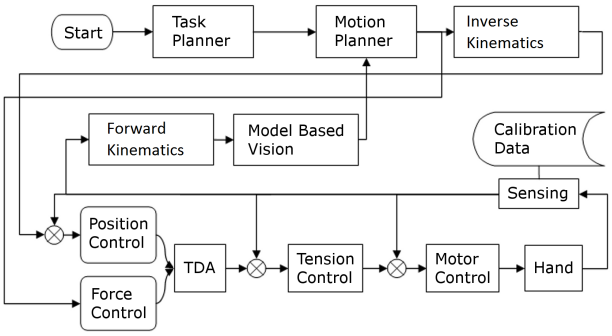


Figure 9.3: Control System Architecture.

motor axes increases and sampling time becomes larger, position control is more appropriate. This produces an over-damped and stable response. If this is the case, then friction and tendon tension compensation terms are not necessary in the tension control algorithm. These characteristics are handled by the motor control boards.

The tension control layer follows. Inputs to this layer are feedback from tendon tension sensors and tendon tension commands. Simple PI control is adequate and simple enough to implement and tune. If the motor control boards operate in position control mode, then the friction and tension compensation elements are not necessary. The tension control layer monitors tendon tensions with the aim of preventing faults. This is done by avoiding applying excessive tensions as well as stopping operation if the tendon tension is below a minimum limit. This is the case when a tendon is loose. In this case, the tendon possibly came off a pulley or is broken.

The tendon distribution algorithm (TDA) is the third layer. Inputs to this layer is the desired force. No feedback is involved since the tendon tensions are controlled in the tension control layer. This algorithm decides the amount of torque to be applied to each tendon to achieve the desired joint torque. The algorithm takes into account maximum and minimum

tendon tensions. A bias force is set and then the tendon tensions deviate equally about the bias force until the difference between them is the desired one. The tendon tensions change following a simple ramp waveform.

Finally, force and position control blocks form the top-level control layer of the robotic hand's hardware. In the case of force control, the force command is fed directly into the TDA stage. If position control is needed, feedback is provided as described in chapter 8. Feedback is provided either by hall effect sensors or by combining the position of the motors with the tendon stretch estimation from the tendon tensions to stretch look up tables (LUTs). The position control block consists of a simple PI controller.

9.6 Conclusions

This chapter presented the design of the actuation system of the metamorphic hand. The metamorphic palm makes it difficult to install gear systems and impossible to install rigid shafts because it is an articulated mechanism. As such, a tendon drive system is used.

Three different control schemes are presented, each with its own advantages and disadvantages. The limitations of a simple PI control scheme are shown. A bang-bang based control scheme which allows easy set-up and reliable operation is presented. Finally, a multi-layer control system is presented and used in the final version of the hand, with all fingers integrated onto the palm.

System telecommunications bus induced delays limit the practicality of centralised control. This is due to the delay introduced by the communication bus. This delay has a degrading effect to the damping ratio of the system and oscillations appear. A greater number of simple controllers for each controlled variable is used. This was preferable to using a central control algorithm to control all variables of the hand.

Depending on the particular implementation of a metamorphic palm-based robotic hand, control system performance may vary. For metamorphic robotic hands with a small number of digits and a fast communications bus, a combination of PI control with friction and tendon stiffness compensation works best. The friction and tendon stiffness compensator compensates a large part of the system parameters. The integral part of the PI controller only has to compensate for the modelling error of the stiffness compensator rather than for all of the tendon stiffness component. For hands with a great number of actuators and reliable but slow communication buses, de-centralising control results in an over-damped system. This responds slowly but remains stable if set-up correctly.

Chapter 10

Sampling Time Analysis

10.1 Introduction

Sampling time is very important when a digital control system is used to control a continuous time plant. Sampling time is good enough when it allows accurate reconstruction of the original signal from the samples of the signal. A rule of thumb regularly used is to have a sampling time 30 times the bandwidth of the measured signal [88, 89]. Control theory focuses on modelling and control of linear time-invarian (LTI) continuous time systems. When the control system is not analogue but digital, both the effect of sampling time as well as quantisation need to be taken into account.

In the case of the KCL metamorphic hand, quantisation is not an issue. All installed sensors have enough resolution to be considered continuous. Sampling time however proved to have adverse effects on the control of the hand. As discussed in chapter 9, if only a few digits are present, centralised PI current control is possible. Ss the number of fingers and motors increase, it becomes impractical to maintain a high enough sampling time. Especially over a

single CANOpen communications bus, as the one used with the hand.

To improve sampling time, a computer networking-inspired approach was used. The structure of CAN bus messages was studied. A routing algorithm was developed to allow building a network of CAN nodes. The nodes are separated over multiple CAN buses. This results in greatly improved sampling time.

10.2 Effects of the Components of the Control System of the Metamorphic Hand on Sampling Time

The sampling time of the control system had to be improved. First each element of the control algorithm is examined and it's effect on sampling time are assessed.

The multi-layer control system of the KCL metamorphic hand requires exchange of various pieces of information among its components. First, data is sampled from the sensors. This is accomplished by communicating with the sensor control boards and requesting the values of the analogue inputs connected to the sensors. This data is then organised in a way appropriate for the processing that follows. Sampling time is also taken into consideration when approximating the time derivatives of the sensed values. The change of a joint position divided by the sampling time yields an estimate of the joint velocity. The sampled data is filtered by means of an finite impulse response (FIR) low-pass filter. This filters out any noise picked up by the sensor and wiring. This kind of filtering also helps improve the stability of the control system. The sensor data is then processed by the main part of the control algorithm. Here is where most of the control signal processing takes place. Normally, a PID controller will process the error data in order to generate appropriate motor control commands. In the case of the hand, a collection of PID and bang-bang controllers with a gain schedule and friction compensation feed-forward

elements are used as described in chapter 9. Finally, the output of the control algorithm is encoded and transmitted to the motor control boards. These control the motors that actuate the robot joints.

A number of test scenarios was designed to assess the real-time performance of each component of the control system. The total sampling time was measured with all components enabled. Tests were performed with only half of the actuators used. This was to assess if the problem scales linearly. Next, all sensors and actuators were disconnected and the control algorithm was let to run with ramp functions as inputs. The output data was discarded. Tests were performed where the sensors were sampled but no control signals were sent to the actuators. Finally, tests were performed with the sensors disabled. Commands were sent to the actuators in such a way so as to prevent them from moving, but still require the same amount of information transmitted as when the system is operating.

Table 10.1: Control System Bottleneck.

Test Conditions	Sampling Time (ms)
Total Sampling Time	93
Half of Motors Connected	47
Motors Disconnected	≤ 1
Only Reading Sensor Data	62
Only Commanding Motors	30

The measured execution times of each iteration of the control software loop, which equals the sampling time, is shown in Tab. 10.1. With all actuators, sensors, and all parts of the control algorithm enabled, the iteration duration was $93ms$. This sampling time was un-

acceptable. It led to oscillations and damages to the tendons transmitting forces from the actuators to the finger joints. During the second round of tests, with half of the actuators and sensors disabled, the iteration duration was on average $47ms$. With all sensors and actuators disabled, the iteration duration of just the software without any communications to the motors was measured. It was found to be $\leq 1ms$. Next, with only the sensors were enabled and no control messages were sent to the actuators. Iteration duration was on average $62ms$. Finally, the sensors were not sampled and only commands to the motors were sent. The iteration duration was $30ms$.

The iteration duration was only $1ms$ when all sensors and actuators were disabled. This indicates the problem is not related to the computational capabilities of the control system. The $1ms$ iteration duration observed during those tests is more than adequate for controlling the hand. The results shown in Tab. 10.1, implied a linear relationship between the sampling time and the number of control nodes. The observation that reading sensor data required double the time compared to sending commands to the actuators led to further investigation of how the low level drivers handled communications to the control nodes. This led to an exploration of the quality of the data link from the main control system to the control nodes and the effect of the number of motors. The latter was expected to exhibit a linear relationship.

10.3 Effect of the Number of Nodes on Sampling Time of the Hand's Control System

Communications from the main control system to each node is handled by a Controller Area Network (CAN) Bus. A number of measurements of sampling time versus number of axes were performed. A linear relationship was expected between the number of nodes and the sampling

time. If a non linear relationship was found, it could imply congestion of the message queues on either the computer's or the motors' side. This would be hard to solve since the size of the queues is fixed and cannot be altered.

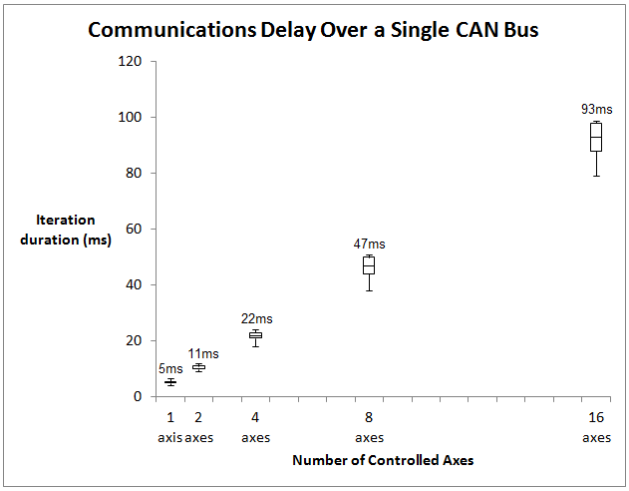


Figure 10.1: CAN Induced Delay.

The test results for the relationship of the number of nodes to the sampling time are shown in Fig. 10.1. The sampling time was recorded for each case. A sample of a hundred data points from each experiment was used. Each bar indicates the median value at it's center. The top and bottom tails of each box indicate the maximum and minimum values of the sampling time. The top and bottom edge of each box indicate the first and third quartiles of the data in each case.

By examining the data, it can be seen there is a linear relationship between the number of nodes and the sampling time of the control system. The distribution of the data in each case is as expected. This is an important observation because it indicates there are no other non-linear phenomena. If for example there was a significant amount of noise interfering with the system, iteration duration would vary widely for each case. If message queues were filling

up then the variance of iteration duration data would significantly increase as the number of axes and sample size is increased. This is because as time went by, messages would be rejected and would have to be resent when there is space available in a message queue. There is some increase in the variance of the data as the number of nodes increases and this is due to more collisions taking place and more re-transmissions needed. The increase in variance is linear which suggests there are no non-linear phenomena such as the overflowing of the message queues.

10.4 Discussion of Sampling Time Analysis for the Hand

Analysis on the effects of the components of the control system and the number of sensors and actuators was performed. This was done by enabling and disabling the components of the control system and varying the number of connected sensors and actuators. The results of the analysis suggested the main contributing factor to the large sampling time is communications delay. These communications included sensor data from the control modules to the main control system and control commands from the main system to the control modules.

These observations led to further examination of the inner workings of the CAN communication protocol. A closer look was taken at the different message types and modes of communication. The structure and content of the transmitted messages was examined. The aim was to identify a more efficient way to communicate with all the sensors and actuators. The transmission line also needed to be probed and examined for signal integrity. Application of computer networking principles is also investigated. Consideration was given to ways to distribute the load in order to reduce the total time required to reach all the nodes on the bus.

First, the signal integrity of the transmission line was assessed. The physical layer of the

CAN protocol is specified having two symmetric signals transmitted on two lines. These are named CAN Hi and CAN Lo. The value of each transmitted bit is inferred by the difference in magnitude between the voltage on the two lines.

Table 10.2: CAN Bus Data States.

Logic	CAN State	CAN Levels
1	Recessive	$CANH - CANL \leq 0.5V$
0	Dominant	$CANH - CANL \geq 0.9V$

Table 10.2 shows the two states of the CAN bus, as well as the voltage difference associated with each state. The first state is entitled "dominant" and is associated with logic level 0. During the dominant state the voltage difference among the two signal lines is larger than $0.9V$. The second state is entitled "recessive" and it corresponds to logic level 1. The recessive state is when the voltage difference of the two lines is smaller than $0.5V$. In the recessive state, both lines are set to high impedance (High Z). In the dominant state, a voltage is applied to both lines but with different polarities.

An oscilloscope was used to probe the transmission line from the main control system to each control node. This is done to verify no significant noise was present on the line and the CAN signals were transmitted as intended. Figure 10.2 shows part of the transmitted signal as viewed on the screen of the oscilloscope. This test was performed at various sampling locations, close to the computer, close to the first motor and at the end of the bus. Some noise was observed but it is negligible. The voltage difference in the dominant state was found to be $2.2V$. The amplitude of the noise was less than $0.2V$. The quality of the edges of the waveform is very good with no overshoot and a flat response in both the dominant and the

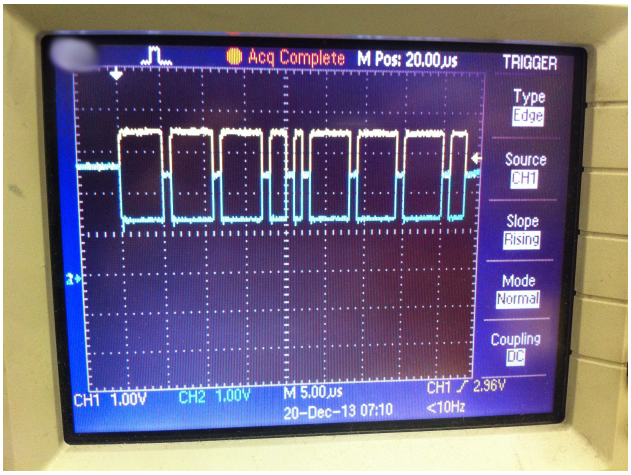


Figure 10.2: Can Signal Integrity Verification.

recessive state.

10.5 Routing for CAN Bus to Reduce the Sampling Time of the Metamorphic Hand Control System

A linear relationship exists among the number of nodes on the bus and the time required to communicate with all those nodes. This is the expected behaviour of a CAN bus based system where all the nodes transmit and receive the same amount of information during each control loop cycle. The preliminary tests indicated that by halving the number of nodes on the bus, sampling time was improved by a factor of 2. By distributing the sixteen nodes of the KCL metamorphic hand into two buses, it was expected that sampling time would drop down to $45ms$ from the current value of $96ms$.

Inspiration was drawn from the way static routing works with a typical local area network (LAN) using the transmission control protocol and internet protocol (TCP/IP) stack. In a

TCP/IP network, each node of the network is assigned an IP address. The IP address is a 32bit number, uniquely identifying each node within the same network. The network consists of network nodes and a number of infrastructure devices, including network routers. The main purpose of the network routers is to act as interfaces between different network segments. These allow distant nodes to communicate by redirecting TCP/IP packets among the network segments the router is connected to. These network segments are also called subnets. In order for the router to know where each message needs to be redirected to, each subnet is assigned an unique address. The subnet address is part of the IP address of each node and it usually corresponds to a number of the most signifficant bits of the IP address. To extract the subnet address from the IP address, a subnet mask is defined. This is a 32bit number with the bits set to logic 1 defining which bits of the IP address of each node correspond to the ID of the subnet. In a simple network with a static routing scheme a technician would designate subnet addresses to each network. The technician would then program static routing tables to each router indicating the network IP address, the subnet mask and the physical interface of the router connected either directly or indirectly through another router to the target subnet.

<table>
<tr><td colspan="17">Header</td><td>Payload</td></tr>
<tr><td colspan="4">Function code</td><td colspan="7">Node ID</td><td>RTR</td><td colspan="5">Length</td><td>Data</td></tr>
<tr><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>1 to 8 Bytes</td></tr>
<tr><td colspan="12">COB-ID</td><td colspan="5"></td><td></td></tr>
</table>

Figure 10.3: CANOpen Frame.

The structure of a CAN Open frame is shown in Fig. 10.3. The first 11 bits of the frame, which are the ID number in the CAN protocol, are split into two sections. The first 4 bits

contain the function code and the remaining 7 contain the CANOpen node ID. These 11 bits are named Communication Object Identifier (COB-ID) in the case of the CANOpen standard. Then follows the remote transfer (RTR) bit. Last is the payload of the frame, which can be from zero to eight bytes of data.

A system similar to how static routing works in TCP/IP networks is used to group the hand's control nodes. The nodes are grouped into two different subnets in order to reduce sampling time by half. Similar to the IP address found at the header of every TCP/IP packet, the header of each CANOpen frame contains the ID number of each node. In order to divide the network into two subnets, only one bit is necessary. The last bit of the node ID number is selected to identify the subnet. This is done so not an entire bus gets arbitration priority over the other. As with TCP/IP there are a number of messages that need to be broadcaster across the whole network. These messages are sent to both subnets when received by the CAN routing algorithm developed for the hand. An example of such a message is the SYNC message that has to be transmitted to all the nodes on the network. The same is true for the messages necessary for the initialisation of the nodes on the network.

Communications with each node is handled by a multi-layer driver. The highest layer of the driver handles the encoding and decoding of the motor control board commands. The next layer handles the generation of the CAN Open messages that are then divided into CAN protocol frames. Next, the CAN frames are sent to the FPGA device of the control system to be queued for transmission. Lastly, the FPGA device sends the queued CAN messages to the appropriate CAN interface module to be transmitted using the physical layer.

Implementation of a CAN router, as described previously, is straightforward. It is desirable to limit the alterations to the driver software as much as possible. To this end, only the FPGA layer of the driver is modified. This is done to hide the fact routing is taking place. The higher

layers of the CAN driver and control software do not need to be changed. When a CAN frame is sent to the FPGA device for transmission, a check is done to its first bit. Depending on the vale of this bit, the CAN frame is forwarded to either physical CAN interface. All CAN messages with a node ID of all zeros, along with some other messages defined by the motor manufacturer, need an explicit routing rule. This allows these messages to be sent to both physical interfaces. Since the last bit of the node ID value is identifying the subnet, all the even numbered nodes are connected to the first bus. All the odd numbered nodes are connected to the second line.

10.6 CANOpen Protocol Message Types and the Metamorphic Hand

As discussed in section 10.2, tests revealed the time needed for the system to read sensor data is twice the time needed to send motor commands. In order to understand why this happens and how to improve sampling time, better understanding of how the CAN Open protocol works in necessary.

The CANOpen protocol uses two different message types for transmitting information. Process Data Objects (PDO) and Service Data Objects (SDO). PDOs differ from SDOs in the way data is transmitted. Data transmission while using SDOs is done by either sending a request and then wait for a response or sending the data and then waiting for an ACK message. PDOs on the other hand are pre-configured so each node knows what data is contained in the transmitted frame. Asynchronous PDO messages are transmitted after an internal or external trigger. An internal trigger could be a threshold value over a measured variable. An external trigger would be the transmission of the same PDO with no data and the RTR flag

set. Synchronous PDO messages however are sent when a SYNC message is received. Each node is pre-configured with all the data it will transmit. Then a single SYNC message is enough to trigger all nodes on the network to transmit the data. This effectively cuts bus utilisation by roughly a half compared to SDOs.

Sampling time is significantly reduced by implementing network routing to use two different transfer lines and then by using PDOs instead of SDOs. Firstly, the sampling time is halved by spreading all the nodes across two subnets. Second, there is another improvement by one third by merging two sensor read messages into one PDO instead of having to send two SDOs. The first reading is the encoder value and one for tendon tension sensor value. Thirdly, there is no need for ACK messages. A single SYNC message is enough to trigger each communication cycle so another halving of sampling time is expected. In total, the final sampling time is expected to be $50\% * 60\% * 50\% = 15\%$ of the sampling time using SDOs and a single transmission line.

10.7 Improved Sampling Time of the Control System of the Metamorphic Hand

An embedded control system was used that contained an FPGA, a real time processor and operating system and a dual CAN module. A CANOpen protocol router was developed for the FPGA so no changes to the RTOS CAN driver were needed. The robot hand should appear as having a single CAN bus to the higher levels.

Figure 10.4 shows the improved sampling rate, stable at 17ms. This value is very close to the expected sampling time of $96ms * 15\% = 14.4ms$ based on the expected improvement enabled by the application of each technique. This change in sampling time had a significant

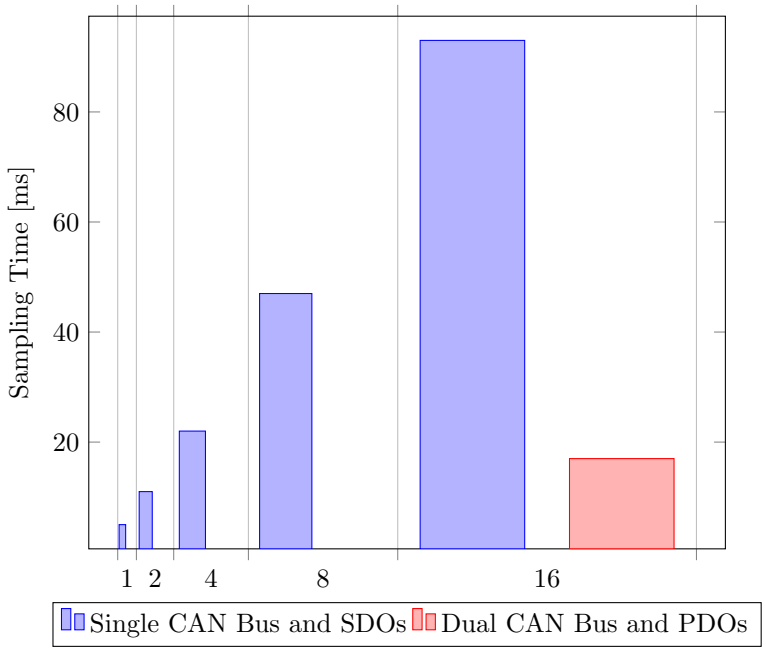


Figure 10.4: Improved Sampling Time.

improvement on set-point tracking for the whole hand.

10.8 Conclusions

The effect of the number of control axis in a CANOpen bus communication system was analysed. Experiments showed a linear relationship in the number of exchanged messages and the communication delay. Based on these observations, a dual-CAN system was developed and a CANOpen routing algorithm was designed. It was found that the choice of the communications bus for a metamorphic hand can greatly impact performance and stability of the system. A complex control system with many nodes had a greatly reduced sampling time, which in turn reduced the control system performance.

By using computer networking-inspired routing methods, a CANOpen routing algorithm

was developed. The new system had a greatly reduced sampling time. This resulted in more reliable control. After implementing all these techniques, a final sampling time of only $17ms$ was observed. This sampling time is very close to the expected value of $14.4ms$. This is 15% of the initial sampling time and significantly improved the performance of the control system of the hand.

These new developments, allowed the construction of a very good and reliable control system for a metamorphic palm-based robotic hand. The control system is capable of both force and position control in all joints.

Chapter 11

Integration of a Metamorphic Hand

11.1 Introduction

The metamorphic palm is a spherical five-bar mechanism designed to be used as a basis for metamorphic robotic hands. By attaching the ground link of the palm to a forearm, and by adding fingers to the other links of the palm, a metamorphic robotic hand is built.

Building a robotic hand with a rigid palm is well studied. The palm merely acts as a holder for all the components of the hand. It also provides a rigid structure for the hand's digits to oppose and mimic the bio-mechanical properties of a real palm. A metamorphic palm however is a mechanism. It is not easy or even possible to install other components inside of the palm, as there is no clearly defined and rigid "inside".

To construct a robotic hand with a metamorphic palm, all hand components must be either in the forearm or in the fingers. In this chapter, the design and assembly of a metamorphic

robotic hand is discussed. Position sensors are integrated into the fingers. A tendon drive system is used to transfer force from the motors located in highly integrated assemblies inside of the hand's forearm.

11.2 Tendon Tension Sensor Slot Design for the Metamorphic Hand

The sensor was mounted into the tendon actuation and sensing assembly. A special slot was designed according to the sensor mechanical limitations.

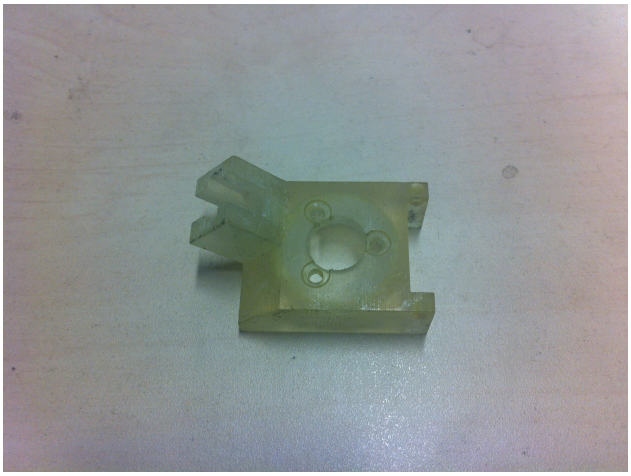


Figure 11.1: Sensor Slot.

The prototype for a single force sensitive resistor is shown in Fig. 11.1. Slot dimensions are dictated by sensor maximum bend and distance from motor gearbox. The part is designed to hold the force sensitive resistor at a 45 degree angle to the face plate of the motor gearbox. This way the tendon-sensor interface is pressed right against the sensor. The effect of friction between the tendon bearing and the walls of the slot is minimised.

11.3 Tension Sensor - Wire Stress Relief Board for the Metamorphic Hand

The sensor is a sensitive device. The sensor's electrical interface comprises a polymer strip bearing two conductive traces. A metal connector is crimped at the end of each trace. There are two possible modes of failure during assembly. First, excessive heat while soldering may damage the sensor. Secondly, we found that if the pins are connected or soldered directly to conductive wires, this will cause excessive stress that will damage the sensor.

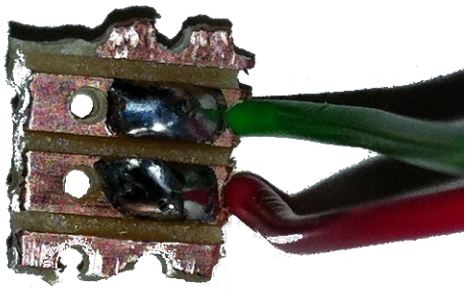


Figure 11.2: Sensor Stress Relief PCB.

To help reduce the stress on the interface of the sensor, a small prototyping pcb was used, as shown in Fig.11.2. The sensor is soldered on one end of the pcb. Copper wires are soldered on the opposite end. This helps in increasing the life of the sensor as well with installation since no extra care has to be taken.

11.4 Tendon Sheath Attachment Point for the Metamorphic Hand

To mount the tendon sheath onto the tendon control assembly, an appropriate slot was designed. The slot was required to have a cylindrical shape for the tendon to go through. The diameter of the cylinder needed to be such to allow an air gap around the tendon to eliminate friction and tendon wear. It also needed to be small enough to prevent the sheath from sliding through the cylindrical gap. Previous prototypes revealed that even small deviation of tolerances in the manufacturing process could result in these requirements not being met. It was also found that a particular kind of hexagonal screw nut had the exact dimensions and strength to hold the outer layer of the sheath in place while allowing small part of the internal Teflon liner through without squeezing it. This led to the decision to design a hexagonal screw slot at the front of the cylinder, as shown in Fig. 11.3.

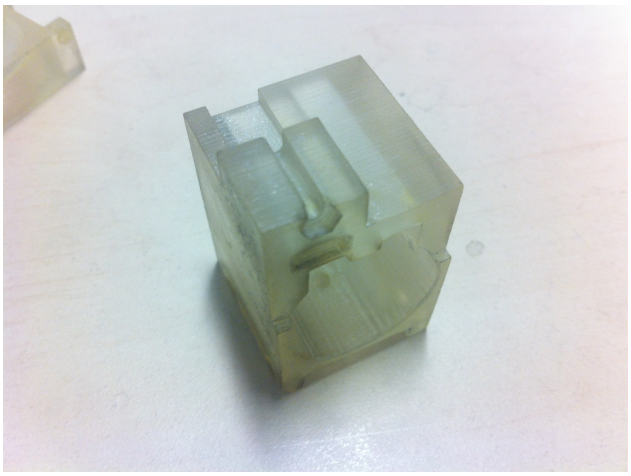


Figure 11.3: Sheath Attachment Point.

This design provides backwards compatibility with previous tendon and tendon sheath

combinations. It also allows the use of a variety of other tendons and sheaths. A different nut can be used or it can be removed completely and have the sheath rest on the nut slot.

11.5 Integrated Motor Socket for the Metamorphic Hand

The new version of the hand, uses considerably more motors than the previous versions to improve finger controllability. This larger number of motors, considerably increased the weight and size of the forearm. To mount the motors in a compact way, a new design for the motor assembly was required. The first option is to mount the motor in-line with the tendon sensing and bearing assembly. This however made the design bulky.

Figure 11.4: Motor.

To help reduce the size and weight of the tendon control assemblies, a slot was designed within the assembly itself. This slot is used to house the motor shown in Fig. 11.4. To decide on the dimensions of the new tendon control assembly, a number of factors were taken into consideration. Motors and gearbox dimensions from the manufacturer CAD drawings were used, after being verified by measuring the existing motors. Further, the minimum thickness of

the additive manufacturing machines used was taken into consideration. Finally, the distance between each individual tendon control assembly was considered. This was to make sure the assemblies could fit in the existing forearm designs.

11.6 Compact Motor-Sensor Assembly Mounting Interface for the Metamorphic Hand

To mount the tendon control assembly into the forearm, screw holes needed to be in place. The design of the mounts of all previous versions of the hand were taken into consideration. Based on the hole placement in those designs and the dimensions of the new design, the hole placement shown in Fig .11.5 was chosen.

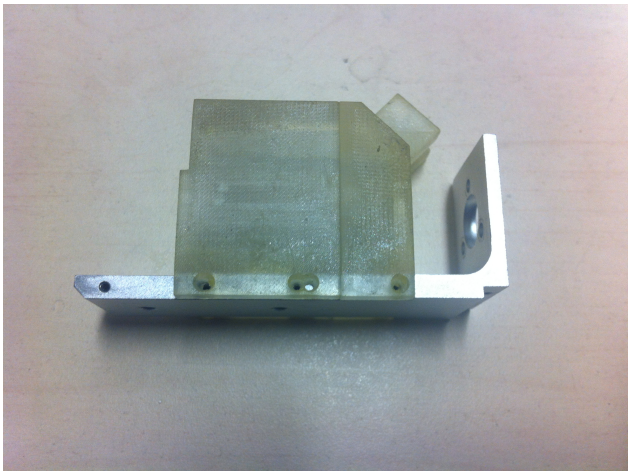


Figure 11.5: Assembly Mounting Interface.

Since the tendon control mounts would be installed next to each other with just 0.5mm tolerance, countersink screws were chosen. This is the most appropriate screw type to help with alignment and prevent screw heads of adjacent tendon control assemblies from touching.

All other types of screw heads would be thicker and thus interfere with the placement of adjacent assemblies. However, extra care is necessary when fastening the screws to not crack the screw holes, made out of common additive manufacturing materials.

Figure 11.6: Tendon-Sensor Interface.

Figure 11.6 shows four tendon actuation and sensing assemblies. These are assembled and mounted in the metamorphic hand forearm. In this figure, the tendon-sensor interface parts are visible. The part that contacts the tendon is formed of a shaft with the option to mount a ball bearing and further decrease friction. The part that contacts the sensor is cylindrical with rounded edges so as to not damage the sensor's surface.

11.7 Motor Assemblies for the Metamorphic Hand

The KCL metamorphic hand is a tendon driven hand. Forces are transmitted from motors to joints through a tendon drive system. Problems common to such systems include tendon breaking as well as tendon relaxation. The effects of tendon breaking are obvious. The tendon breaks and the result is usually catastrophic. Tendon loosening however is not a binary

phenomenon and can have various effects. The worst case scenario would be the tendon coming off a pulley. This significantly alters the operating parameters of the system. The system will at best experience degraded performance. At worst it will become unstable. Moreover, the tendon that is now moving freely might get entangled to another component of the robot and cause further damage. The tendon will eventually break since instead of sliding on a pulley, it is sliding on some surface not designed for that purpose and damages the tendon each time it is moved.

A number of tendon layouts and motor assembly designs have been tested. Initially, a design with two tendons on the same motor was used. This is the first design from the left on fig. 11.8. Two pulleys were attached to the motor shaft. Each pulley carried a tendon attached in a manner such that when the motor rotated, one tendon was pulled and the other was relaxed. Such a design simplifies the control system and can have very good performance. In fact, it is used in a number of tendon driven hands already deployed in the industry. Such a design however is very sensitive to manufacturing tolerances. Further, it does not provide for precise control of the joint torque since this becomes directly related to the stiffness of each tendon, the friction of the transmission system and the error in the position of the joint due to external disturbances. Furthermore, the force applied to the tendons by the installed pre-tension springs introduces significant backlash. If this force is to be increased so as to decrease backlash, the maximum force that can be transmitted over the system is reduced. This is because the tendons have a breaking point and the bias force is now larger.

With this design decision in mind, designing the new motor assemblies was a straightforward process. A single spool was now needed on the motor shaft. There was no need for the pre-tension assembly since the tendon tension could be directly monitored. This led to the design shown in fig. 11.7. The pulley rests on a pressure sensitive resistor. The interface

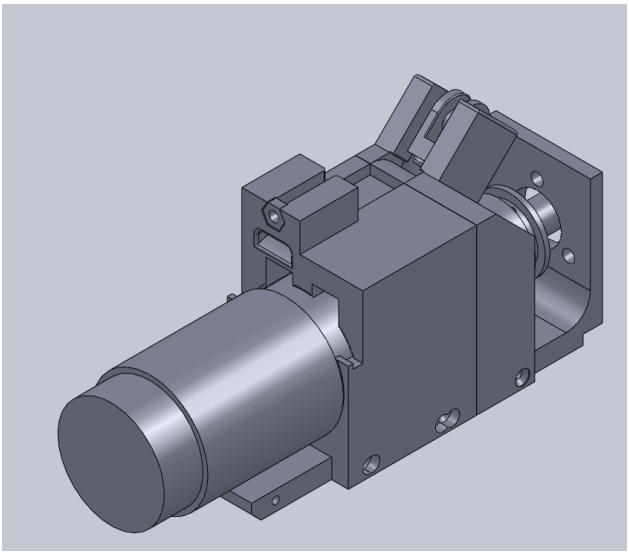


Figure 11.7: Motor Assembly Design.

between the two is designed to prevent damage to the sensor. A piece of elastic can also be used to improve the quality of the measurement and further increase the life time of the components. The tendon is attached to a single pulley on the motor. The sensor slot and tendon-sensor interface are at the rightmost part of the figure. Attached to them is the sheath attachment point. The motor is secured at the centre of the assembly.

This new design has the added benefit of making replacing pulleys easier by increasing their accessibility to the user. The evolution of the motor assemblies is shown in fig. 11.8. On the left is the first generation motor assembly. It is followed by the second generation assembly at the centre. Since these versions were designed to house two tendons, they needed to have a tendon tensioning assembly to function properly. The new assembly is on the right. Because each assembly controls a single tendon, there is no need for the pre-tension component. Notice the pressure sensitive resistor strip on the left side of the tendon mounting interface.

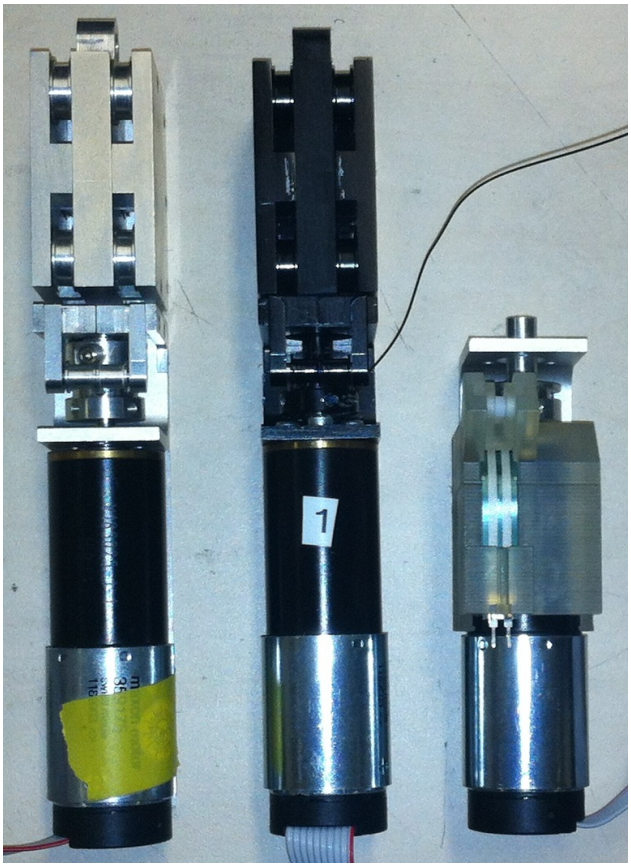


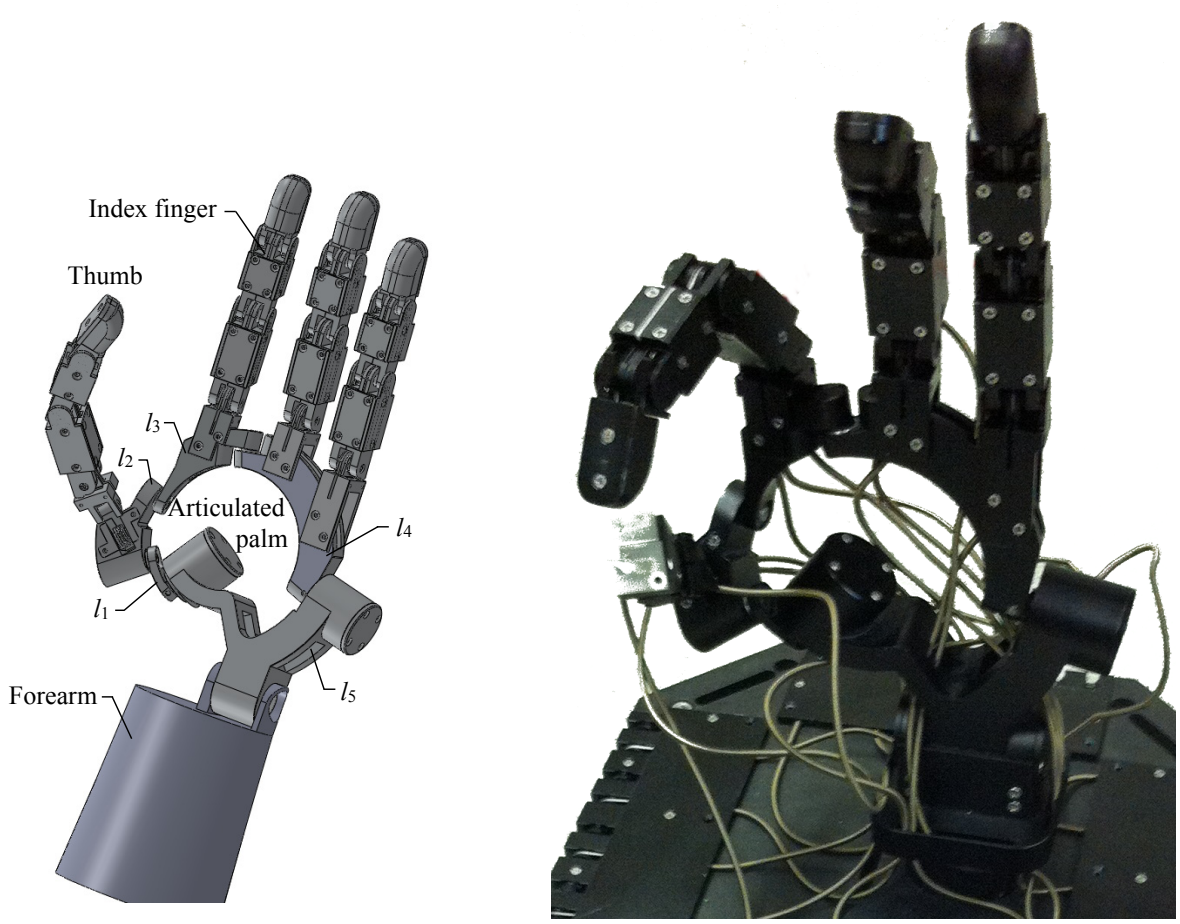
Figure 11.8: Motor Assembly Evolution.

11.8 Real-Time Control Hardware of the Metamorphic Hand

To improve disturbance rejection and simplify the process of tuning sixteen controllers, the control logic was split into two stages. The first stage is a simple P controller, which commands a change in motor position based on tension error. The second stage is a commercially available board provided by the motor manufacturer. Tuning this second stage was done by the in-built auto-tuning algorithm. Tuning of the first stage was done through trial and error.

11.9 Structure and Geometry of a Multi-Fingered Metamorphic Robotic Hand

The new robotic hand, based on the concept of metamorphosis, is presented in Fig. 11.9a.



(a) A Multi-fingered Metamorphic Robotic Hand. (b) KCL Metamorphic Hand Physical Prototype.

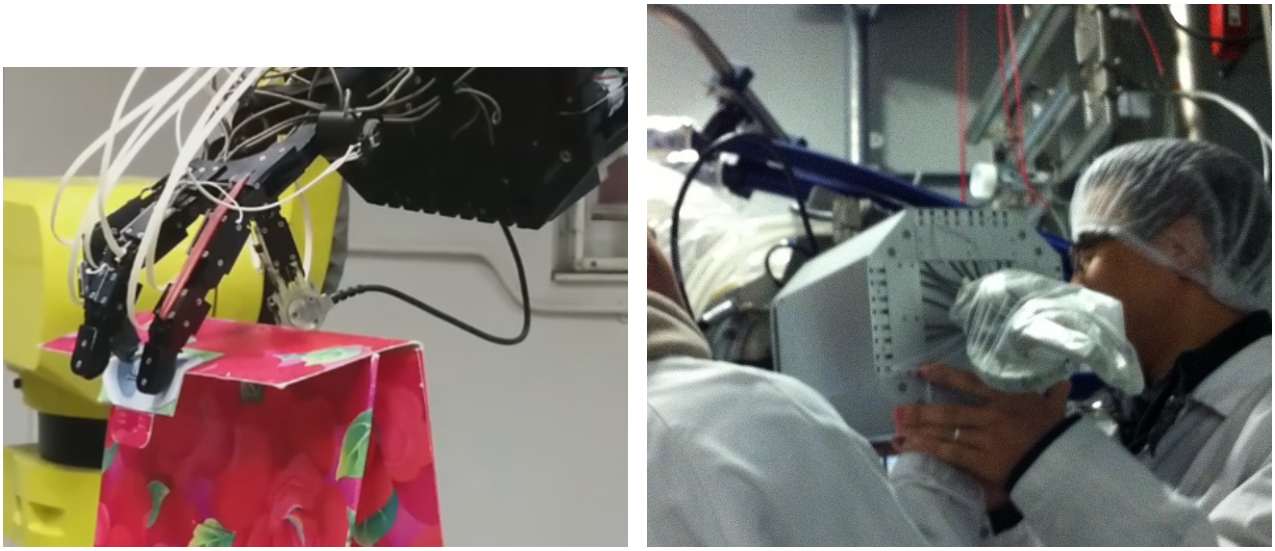
Figure 11.9: KCL Metamorphic Hand.

Figure 11.9a illustrates the structure of the new multi-fingered metamorphic robotic hand with an articulated palm by Wei et al. Figure 11.9b shows the practical implementation of that design. The palm is formed by a spherical five-bar linkage comprising five links, The links

are denoted as l_1, l_2, l_3, l_4 and l_5 with the base link l_5 connected to a wrist that is linked to the forearm. All the fingers are mounted to the links of the articulated palm. The thumb is mounted to link l_2. The index finger is mounted to link l_3. The rest of the fingers are mounted on link l_4. In this robotic hand, except for the thumb, each of the fingers contain only three revolute joints with parallel axis of rotation. These joints allow only flexion/extension motions but no adduction/abduction motion. The introduction of the articulated palm compensates for the absence of adduction/abduction motions of the fingers. The palm enables dexterous manipulation and grasping capabilities by adapting the configuration of the hand for various tasks and different environments.

In order to increase the dexterity of the palm, a number of factors are taken into consideration during the design phase. Human hand structure considerations contributed anthropomorphic elements in the design. The rotatability criterion of the spherical linkage [81] was used in deciding on the kinematics of the hand. The singularity avoidance design criteria presented in this dissertation played an important role. The angles of the links satisfy $\alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 + \alpha_5 = 360^\circ$ and $\alpha_1 + \alpha_2 + \alpha_5 > \alpha_3 + \alpha_4$. The palm itself has two degrees of freedom such that two drives are used to adjust configurations of the palm. In particular, The drive adjacent to link l_1 is used to change the structure of the articulated palm. This is accomplished by rotating the crank link, i.e. link l_1, to form a four-bar linkage in instant and innate metamorphic phases [37].

A practical configuration of the hand for an origami carton folding experiment carried out during the EU project TOMSY is shown in Fig. 11.10a. Visible are the sensor wires as they exit the top of the fingers. The tendon sheaths connected at the base of each finger on the palm are also also visible. An origami type crush-lock carton is being folded to demonstrate the dexterity of the metamorphic robotic hand.



(a) Origami Carton Folding Experiment.

(b) DEXterous DEBoning Experiment.

Figure 11.10: Metamorphic Hand Physical Prototypes.

Figure 11.10b shows the hand fitted to a robotic arm for a dexterous de-boning experiment. A human worker holds a knife on one hand and has a metal glove on the other hand for protection. The purpose of the experiment was for the hand-arm system to assist the butcher during the de-boning operation. The hand is dressed with a flexible latex glove in compliance with meat handling regulations. The glove also serves to protect the hand from the moisture and liquids present in the piece of beef being handled. In addition, the glove provides a surface which the fingers can oppose and thus more accurately replicate the bio-mechanical characteristics of a human palm.

11.10 Conclusions

This chapter presented the integration of the KCL metamorphic hand. The goal is to use the metamorphic palm as a basis for a robotic hand. All the necessary components required to

form a robotic hand are attached onto the palm.

The design of each component of the hand is presented. The mechanical interfaces between components are shown. The tendon sensing and actuation assemblies evolved to allow for better sensing and control of the tendon tension in a more compact form. The new design also allows for the tendon sheaths to be mounted directly on the assembly for use in devices not incorporating tendon sheath anchor points.

It is very important to consider final integration of the system right from the start of the design process. Decisions made at the early design stages have to take into account the mechanical limitations and mechanical interfaces of all the components involved. The tendon sensing and actuation assemblies were designed while taking into account the dimensions of the existing components and in-house manufacturing capabilities. Manufacturing process variation is an important factor and was considered in all mechanical interfaces between different components. The design allowed for larger gaps between mating surfaces. Secondly, the types of screws used was carefully considered. Countersunk screws where possible where possible, taking care not to over-tighten them during assembly and risk splitting screw holes.

The new tendon sensing and actuation assemblies were designed with compactness in mind. They are compatible with the existing forearm design and can accommodate the same motors and tendons. The new assemblies are half the length with added sensing functionality. They can also be used stand-alone, with no need for special tendon sheath anchor points on the structure they are attached to. In fact, they do not require any kind of special structure since the tendon sheaths can be anchored directly onto the sensing and actuation assemblies. Further, the new assemblies contain a single moving part, the sensor-tendon mechanical interface, which may or may not require a tendon bearing, depending on the used materials and involved tensions.

Following all these considerations, a metamorphic palm-based robotic hand was built, capable of dexterous manipulation of a variety of rigid and articulated objects. The hand is capable of force and position control. The palm of the hand can also be covered by rigid plates to better mimic the bio-mechanical properties of a human palm if necessary.

Chapter 12

Carton Folding Operation

12.1 Introduction

An articulated object is manipulated to demonstrate the dexterity of a robotic hand with a metamorphic palm. The articulated object is an origami-style packaging carton.

A standard robotic gripper with a rigid palm is not dexterous enough to perform the delicate operations required to fold such an object. Purpose built origami folding mechanisms are efficient but can only fold a limited number of origami designs.

By using the KCL metamorphic hand, it was possible to accurately follow the folding instructions of a human expert. Tele-operating the robotic hand was simple, thanks to the control system shown in chapter 9. The visualisation of the robotic hand shown in chapter 6 was used to study the hand prior to the folding operation. Doing so provided insights in the hand's capabilities in the context of this experiment.

12.2 Carton Folding Task Description

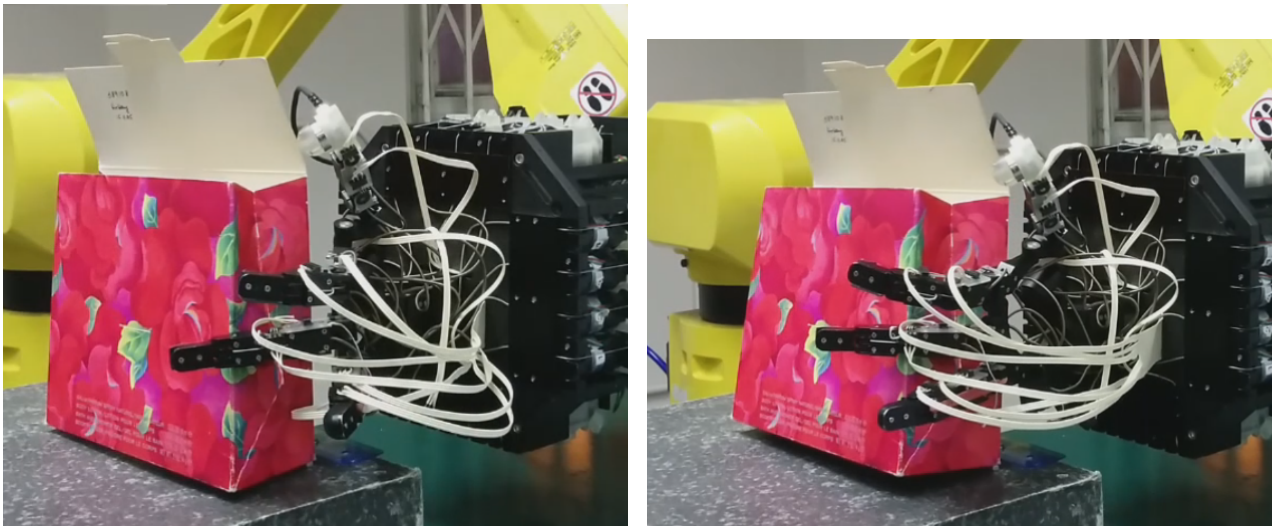
The carton folding demonstration for this particular carton comprised five distinct phases.

1. Locking the carton.
2. Testing the correct lock.
3. Complete locking of side panel.
4. Folding the top panel.
5. Tucking in the flocking flap of the top panel.

12.3 Locking The Base of the Carton

The base of the carton is a parallel mechanism. It is designed to become start in a flat configuration and lock into a rectangular shape when pushed from the side. In order to lock the carton, the back panel of the carton is held into place by a metal peg. The front panel is pushed until the crush-lock mechanism at the base locks. This is the first and most important step in the folding process. A bad lock will lead to a failed folding attempt and a possibly damaged carton.

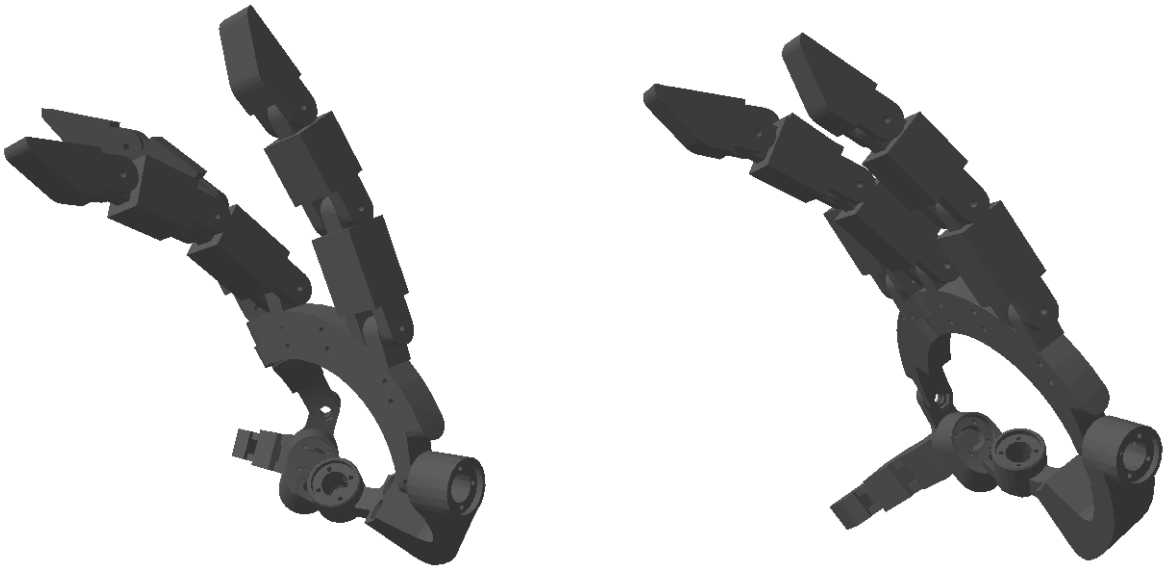
Figure 12.1 shows the hand operating the carton in order to lock it. The back panel is held in place by a metal peg. This is so that the carton does not move during the operation. The hand is positioned so a force can be applied parallel to the front panel. This causes the parallelogram mechanism formed by the front, back, and side panels to become orthogonal. This forces the panels of the crush-lock mechanism, the base of the carton, to become co-planar and lock.



(a) Before Locking.

(b) After Locking.

Figure 12.1: Carton Locking Phase



(a) Before Locking, Palm.

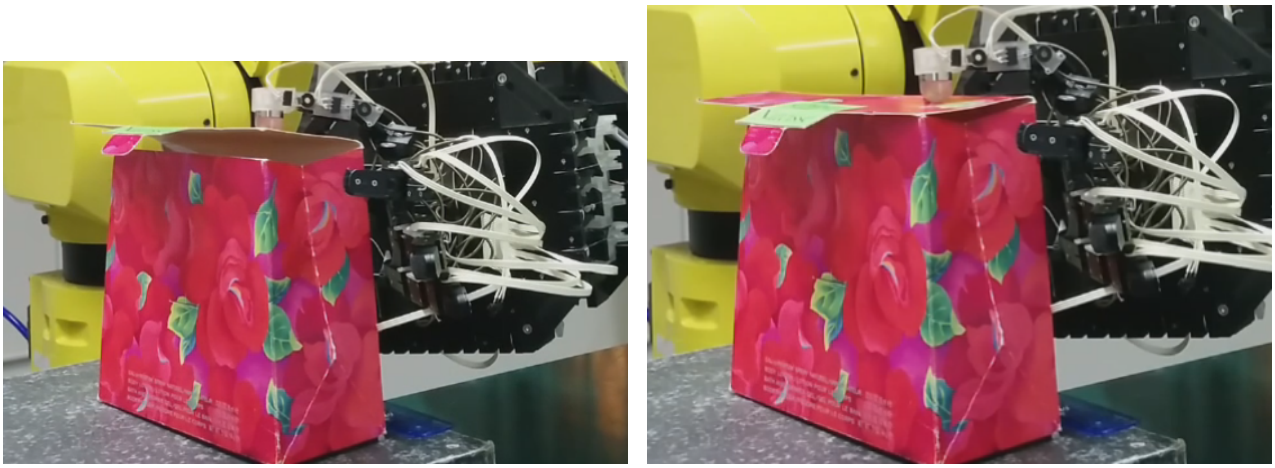
(b) After Locking, Palm.

Figure 12.2: Carton Locking Phase, Palm

Figure 12.2 clearly shows the configuration of the palm during this step. The applied force is largely due to the actuator connected to the rocker link. This actuator moves the middle and index fingers to push and lock the carton. The front panel is pushed by the proximal interphalangeal joint of the middle finger.

12.4 Checking the Correct Locking of the Base of the Carton

The next step in the folding process is to check the correct locking of the carton. The side panels have two possible states. They can either be in the convex or concave configuration. In the convex configuration, the side panels are bent inwards and the carton can be folded correctly. In the concave configuration, the side panels are bent outwards and as a result the carton is not folded properly.

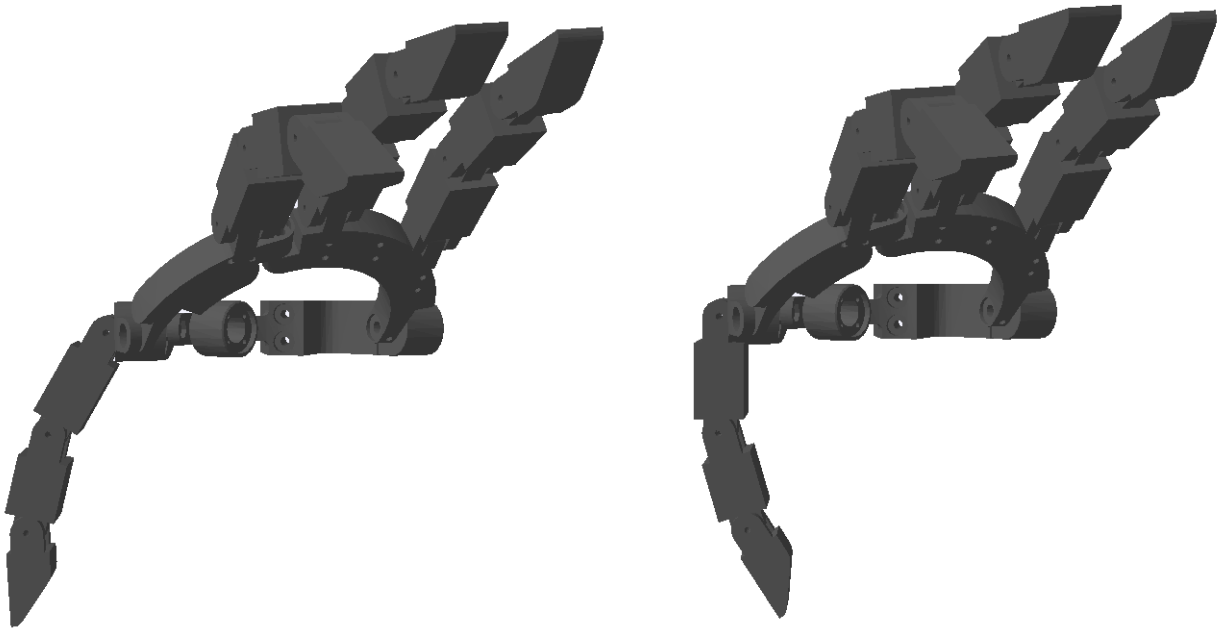


(a) CAD Drawing, Coupler Non-Reflex Configuration. (b) CAD Drawing, Coupler Reflex Configuration.

Figure 12.3: Coupler Non-Reflex and Reflex Configurations

The thumb is positioned to push the top panel downwards, as shown in Fig. 12.3. The

goal of this step is to position the thumb in such a way so it can push down on the top flap with an integrated force-torque sensor. This is to assess the correct execution of the locking task. If the carton is locked correctly, the top edges of the back and front panels should be in contact with each other and a very small force is enough to fold the top panel. In the even the carton is not locked properly, the body of the carton forms a rectangular box. In that case, the top flap can not be fully folded. The side panel that is in the concave configuration needs to be pushed to be changed to the convex configuration.



(a) Before Locking, Palm.

(b) After Locking, Palm.

Figure 12.4: Carton Locking Phase, Palm

Figure Fig. 12.4 shows the palm configuration used in this task. The palm is configured so the operation plane of the thumb is normal to the joint formed by the top and back panels. This way, the trajectory of the thumb is very simple and only involves rotating the MCP joint of the thumb. The forces at the thumb tip are recorded. If during this operation the force

exceeds a threshold value while the top flap is not folded correctly, this indicates that one of the side panels is in the wrong configuration.

12.5 Side Panel Concave to Convex Configuration Change

The side panels of the carton have two possbile configurations, convex or concave. The carton is designed so when folded, the side panels must be in the convex configuration. The carton is pre-creased so by folding of the side panels, the result is a convex fold. If one of the side panels is in the concave configuration, the carton has a tendency to form a rectangular box instead of a wedge shape. After determining the configuration of the side panel, if it is the wrong configuration, the panel is pushed so it switches to the convex configuration.

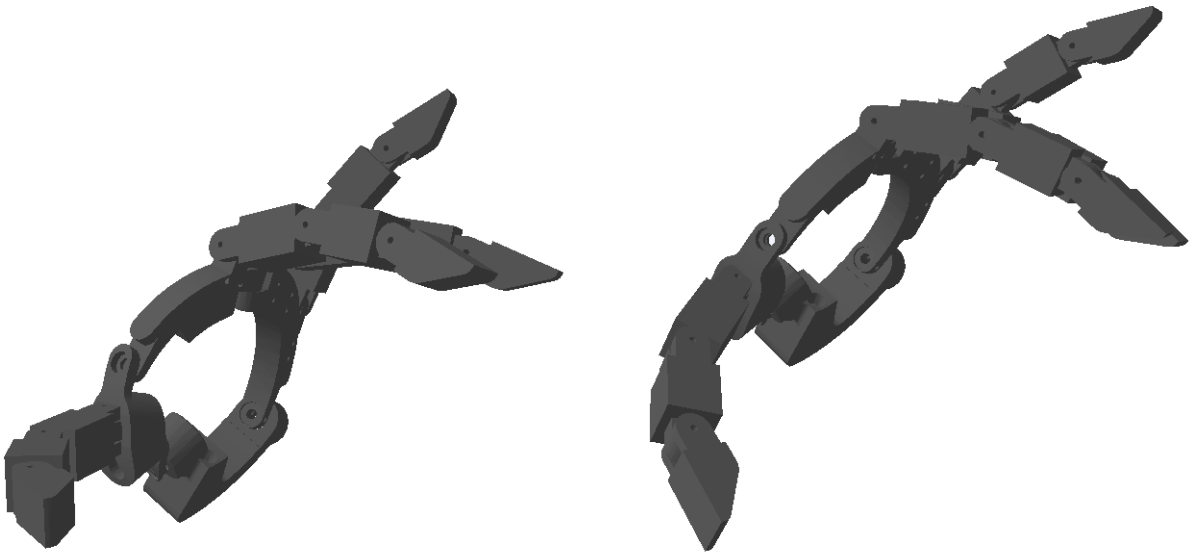
(a) Side Panel Concave Configuration.

(b) Carton Side Panel Convex Configuration.

Figure 12.5: Coupler Non-Reflex and Reflex Configurations

Figure 12.5 shows the hand pushing the side panel to change its configuration from concave to convex. The side panel can be thought of as a bistable mechanism at this step of the carton folding task. In order to transition from one configuration to the other, the center of the mechanism has to be pushed and reach the flat and singular configuration. It then has to be

pushed some more to transition to the convex configuration. The force required for such an action can not be provided by only one serial chain, formed by one of the palm joints and a finger. The metamorphic palm is now set to the coupler-joint straight configuration. In this configuration, the rocker link actuator transfers a force to the rocker-mounted fingers. At the same time force is transferred to the rocker link through the coupler links from the the crank link actuator. The combination of the forces produced by both actuators is now enough to change the configuration of the side panel to convex.



(a) Before Locking, Palm. (b) After Locking, Palm.

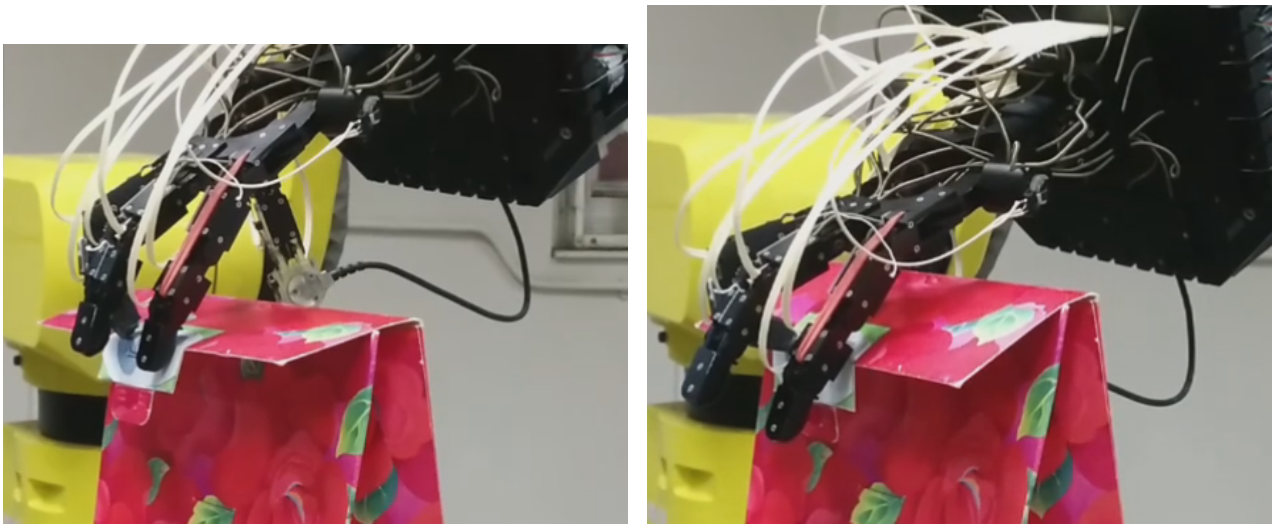
Figure 12.6: Carton Locking Phase, Palm

Figure 12.6 shows in better detail the configuration of the palm during the pushing of the side panel. The singularity of the coupler joint is clear. The mechanism is effectively reduced to a spherical four-bar mechanism. The forces from both actuators are transferred to the finger performing the push. The hand dexterity is reduced since the introduction of this joint angle value constraint for the joint-coupler angle. This is a good example of the use of

the metamorphic palm to adapt the hand. Hand dexterity is traded for strength in order to perform a subtask that requires a simple motion but large forces.

12.6 Folding the Top Panel

Following the successful lock of the base and correct folding of the side panels, the next step is to fold the top flap of the carton. The top flap needs to be pushed downwards and towards the front panel. The top panel, as with all other panels, is pre-creased. A relatively small force is required to fold it. Even so, while the panel is pushed down and towards the front panel, the carton may fall backwards. The thumb is used to support the carton during this folding operation.



(a) CAD Drawing, Coupler Non-Reflex Configuration. (b) CAD Drawing, Coupler Reflex Configuration.

Figure 12.7: Coupler Non-Reflex and Reflex Configurations

Figure 12.7 shows the hand positioned to fold the top panel. At the start of the task, the palm is in a singular configuration to assist the palm actuators overcoming gravity. When

contact is made, the palm is configured to allow the fingers to fold the top panel downwards. The thumb is positioned to support the carton from the back during this folding step. Folding is accomplished by the combined action of the fingers and palm. The small, middle and index fingers and the palm body opposite the rocker link, push the top flap downwards. The thumb is placed tangent to the back panel to prevent the carton from tipping over.

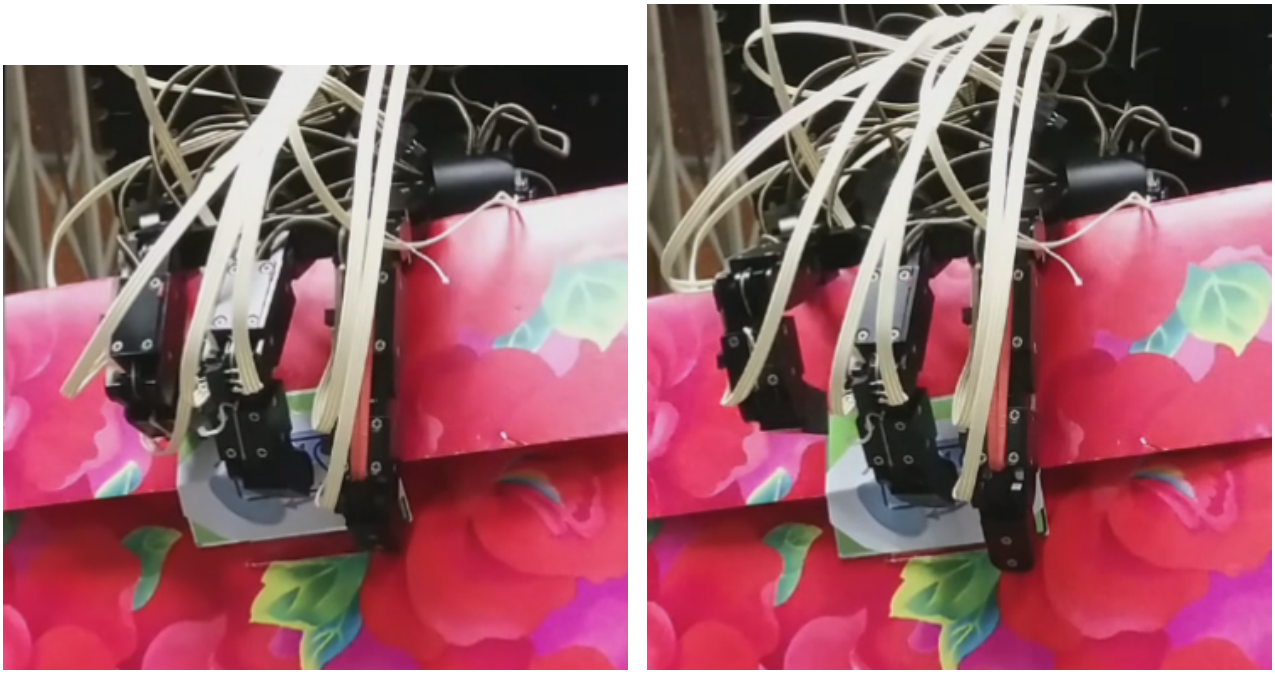
Figure 12.8: Before Locking, Palm.

Figure 12.8 shows the visualisation of the hand during the top panel folding operation. The configurarion of the palm during the approach of the carton from the top is clearly shown. This configuration allows the hand to resist gravitational forces and support its own weight. The palm is configured so the thumb can support the back panel of the carton and still use the rocker link and enable the fingers to fold the top flap.

12.7 Tucking In Locking Flap of the Top Panel

After folding, the top panel has to be locked in place or else the carton will not stay closed. The top panel has a small locking flap. It is a small flap designed to slot into a cut on the

surface of the front panel. In order to insert the locking flap into the slot of the front panel, the top panel and flap need to be bent. This bending has to be small in order to not crack the virtual joint formed by the edge of the top panel on the side of the locking flap. If the bending force is large then a crease will form and the carton will not lock closed.



(a) CAD Drawing, Coupler Non-Reflex Configuration. (b) CAD Drawing, Coupler Reflex Configuration.

Figure 12.9: Coupler Non-Reflex and Reflex Configurations

The tucking in operation, required to lock the top panel of the carton, is shown in Fig. 12.9. First, the fingers are positioned so as to bend the locking flap inwards and facilitate the tucking action. This is accomplished by the little finger fully bending in order to control the top flap. Then the middle finger makes contact with the center of the locking flap. The index finger grabs the locking flap from the side of the locking edge in order to bend it. While the fingers maintain their posture, the rocker link of the palm is actuated to rotate the top panel. This causes the locking edge of the flap to make contact with the front panel. Finally, the palm

keeps folding the top panel while the fingers gradually reduce the forces they are applying. This makes the locking flap slide into the locking slot of the front panel.

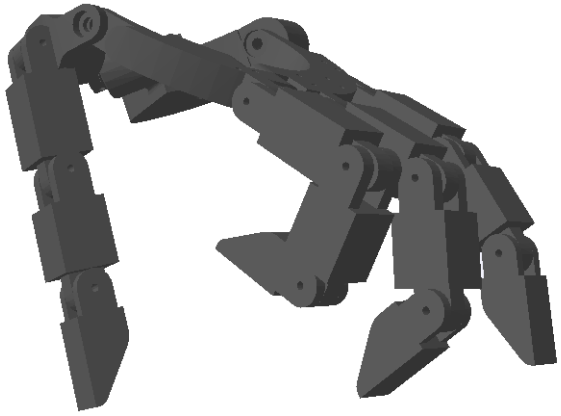


Figure 12.10: Before Locking, Palm.

Figure 12.10 shows the visualisation of the palm during the locking flap tucking operation. The palm is configured to allow the thumb to support the carton from the back. At the same time, the rocker link of the palm allows the fingers to operate the locking flap. The rocker link of the palm is used to apply a force to initially maintain the fold of the top panel and finally complete the tucking operation. The tucking operation is completed by completely folding the top panel and sliding the locking flap into the front panel slot.

12.8 Conclusions

This chapter presented the carton folding operation using the KCL metamorphic hand. As demonstrated, the metamorphic palm enhances the dexterity of the hand and allows execution of complex tasks such as folding of origami cartons. The carton folding task is broken down into simple subtasks and the hand is tele-operated. The operator sends force commands to the controller system of the hand. Then, the tension of each tendon is individually controlled to

achieve the desired forces and motion. A simulation of the hand at the start and end of each subtask is provided to better understand the motion in each part of the folding operation.

During the carton folding task, the need to alternate from position control to force control and back to position control was identified. At the start of each subtask the fingers need to be positioned at the right points over the panels of the carton. During the manipulation phase of each subtask it is important to control force and direction of motion rather than joint position. This way the panels can be folded without damaging the panels or the hand tendons. The position command of each finger acts as a limit to the motion rather than a set point.

By using spherical trigonometry-constrained kinematics, which can differentiate joint-coupler reflex and non-reflex configurations, a real-time simulation of the hand is possible. This simulation helps the operator visualise the hand before committing to a particular course of action. Although the final control system did not include the visualisation element, visualisation was used to experiment with a virtual hand and identify potentially useful configurations before using the real hand. The carton folding experiment is a good demonstration of the versatility and dexterity of the metamorphic hand. By changing the configuration of the metamorphic palm, the hand can adapt to different task requirements. From subtasks requiring greater strength than dexterity to subtasks demanding more dexterity than strength.

Chapter 13

Conclusions

13.1 Introduction

This dissertation explored how to model and control metamorphic mechanisms in the context of metamorphic manipulators and metamorphic hands. Metamorphic mechanisms have the ability to adapt to changing task requirements. As such they can grant this ability to manipulators and hands evolved from metamorphic mechanism designs. The following were the two main questions this study set out to answer:

1 How can the kinematics of spherical linkage-based metamorphic mechanisms, namely the vA joint and metamorphic palm palm be solved, in a way useful for simulation and control, that does not fail in the case of a singular configuration.

2 How to develop a control system capable of controlling metamorphic hands that is reliable, enables position and force control, and can scale easily from a small number of joints for simpler mechanisms to a larger number of joints for dexterous operations.

13.2 Theoretical Findings and New Developments

The first question concerned the theoretical analysis and simulation of the vA joint and metamorphic palm and of a metamorphic palm evolved hand. These mechanisms were analysed both in the joint space and in the workspace. The spherical nature of these mechanisms pointed towards spherical trigonometry.

The exploration of the usefulness of spherical trigonometry in metamorphic mechanism modelling and control started with the variable axis joint, as discussed in chapter 3. It was demonstrated spherical trigonometry is well suited to form a basis for metamorphosis trajectory planning. This is attributed primarily to spherical trigonometry's intrinsic spherical motion constraints.

The study continued in chapter 4 where a method was presented on how to segregate the spherical palm into a set of spherical triangles and enable the further analysis of the mechanism through spherical trigonometry. Although there are previous works on the analysis of spherical mechanisms, the need was arisen to differentiate the joint-coupler reflex and non-reflex configurations. This new method fully solves the metamorphic palm and enables easy singularity analysis and simulation.

Thirdly, based on the above results, singularity analysis of the metamorphic palm was performed in chapter 5. Through the use of spherical trigonometry-constrained kinematics, all singular configurations were identified. A set of singularity avoidance-based design criteria was presented and a solution for the singular but controllable cases was derived.

Finally, by combining these results, a joint-space simulation of the metamorphic palm along with a metamorphic hand design and its workspace simulation were presented in chapter 6. It was show how the metamorphic palm can be used as a basis for a metamorphic hand and how it enhances the workspace of the hand. Further, a number of numerical examples and

possible grasps were presented.

The major contributions in this dissertation are published as follows:

- The work relating to the variable axis joint metamorphosis is published in [90].
- The spherical trigonometry-based kinematics work is published in [91].
- Contributions related to the hand control system for meat deboning operations are published in [92].
- Finally, early work on the vA joint-based parallel manipulator is part of publication [34].

13.3 Empirical Findings and New Developments

The second part of this dissertation focuses on the development of a new reliable and scalable control system for metamorphic palm-based dexterous robotic hands.

Following the theoretical developments, experimental work commenced using existing hand prototypes. First, as presented in chapter 7, an analysis of the existing hands was conducted. A number of failure modes were identified and an experimental platform was designed and built to facilitate experiments and data collection.

Using the above mentioned observations and experimental platform, the study progressed by exploring a number of tendon tension and joint position sensing schemes as presented in chapter 8. A sensor-less scheme was developed for cases where the built and assembly quality is of high standard and the forces exerted by the joints are small. An alternative sensing system which requires sensors in all joints was finally used for manipulating heavier and stiffer objects.

The study then progressed with actuation, as presented in chapter 9. The limitations of the most popular PI control scheme for actuating tendon-driven metamorphic hands were

identified. Based on these observations, two different control schemes are presented. One scheme has better performance and is best suited for low degrees of freedom hands. The other scheme has better stability and is aimed at hands with more points of articulation.

While developing the control system for a dexterous hand with a high number of degrees of freedom, the effect of the number of joint axes on sampling time became significant. The importance of sampling time is discussed in chapter 10. A method was developed to increase the sampling frequency in CANOpen-based hands, which can potentially be applied to any bus-based system with a unique address for each control node.

Chapter 11 provides the details of integrating all the theoretical and technical developments into a new control system for a metamorphic hand. New motor assemblies were designed that incorporate the new sensing and actuation scheme. The theoretical developments were used to develop a real-time simulation and visualisation of the hand. These were integrated into the new control software, for the user to look at the hand's pose before committing the commands to the motors.

Finally, chapter 12 presented a carton folding operation. In this experiment, the metamorphic hand with the new control system is used to fold a complex origami-type packaging carton. The metamorphic palm-enhanced dexterity of the hand and the capabilities of the new control system are demonstrated. Photos of the operation are provided along with spherical trigonometry-constrained kinematics-produced simulations to better understand the underlying operation in each subtask.

13.4 Theoretical Implications

The majority of existing robotic hands are built with a fixed palm. In this dissertation, it is shown an articulated palm enhances the workspace and the ability of a robotic hand to

manipulate complex, articulated objects. The kinematics and control systems for these types of hands however pose new challenges, not found in conventional hand designs.

First, while it is possible to solve the kinematics of such mechanisms using generalised methodologies, like loop equations, the D-H method, quaternions, or even screw theory, the spherical trigonometry-based solution presented in this dissertation involves simpler equations. This solution also forms a good basis for identifying the singularities of the palm mechanism. It also simplifies the identification and differentiation of the joint-coupler reflex and non-reflex configurations. The spherical trigonometry-based results are also simple to implement in a computer program for simulation and visualisation purposes.

Second, the majority of tendon driven robotic hands that employ tendons for the transmission of motion, incorporate fixed and rigid tendon paths for all the tendons. In these cases, accurate tendon models can be made and characterised. In the case of metamorphic hands however, having fixed paths for the tendons that actuate the fingers is challenging. Two solutions were presented. For the case of a simpler hand with fast sampling time, it was shown that using a PID controller and compensating for friction and tendon tension yielded good tension tracking results. For hands with many degrees of freedom and slow control loops, it is best to use a distributed and over-damped system with the most sensitive parts implemented in very fast hardware and very close to the actuators.

13.5 Recommendations for Future Research

In the context of the variable axis joint and the metamorphic parallel manipulator, future work could include integrating the proposed algorithm into the control system of the metamorphic parallel manipulator and assessing its performance in various tasks. Further, static and dynamic analysis could reveal new uses for the vA joint, to minimise actuation effort by

aligning the joint plane in the case of an R_e configuration with the limbs of the mechanism.

In the context of the metamorphic palm and hand, mechanism dynamics were not explored in this study. One potential future research topic is to identify ways to use these spherical trigonometry-constrained kinematics in conjunction with standard methods for mechanism dynamics, such as Newton-Euler or Screw theory.

Another point for future research is to analyse mechanism statics and further explore metamorphic palm configurations that maximise applied forces by the hand. During the carton folding operation it was observed in some configurations it was possible to use both palm actuators to push panels of the carton. The panel would otherwise not move when using only the fingers. The ability of metamorphic hands to trade dexterity for strength, by adding motion constraints in the form of an optimal trajectory, might be promising.

Force transmission from the actuators to the joints is a point that could be improved significantly. Flexible, free-standing tendons have very hard to model behaviour and their parameters change significantly. Especially after a change in the shape of the tendon sheath, resulting from a change in the palm configuration. Identifying and developing ways to improve the transmission system, for example by somehow fixing the tendon paths or by using linkages, gears, or other means is required to improve the performance of metamorphic robotic hands.

Motion planning algorithms for grasping and manipulation is another field that needs to be addressed. It would be interesting to see how existing path planing algorithms can be applied on the point clouds generated during the joint space analysis of the mechanism. New algorithms might need to be developed specifically for metamorphic hands.

13.6 Conclusions

Although spherical mechanisms are well studied, not much focus is in the direction of using spherical linkages as bases for metamorphic mechanisms. In this study, it was shown spherical trigonometry is very useful in analysing spherical linkage-based metamorphic mechanisms. Spherical trigonometry-constrained kinematics not only can be used to perform joint space and workspace analysis and simulation, but to also identify singularities just by inspection. Spherical trigonometry also provides elegant tools for trajectory planning for reconfiguring spherical mechanisms. This is to be expected as spherical trigonometry is intrinsically constrained to the very nature of spherical mechanisms, contrary to more general methods for solving kinematics, such as the D-H method, loop equations, or screw theory. While spherical trigonometry obviously falls short in more general mechanisms with spacial motion while these methods can be applied to any kind of mechanism, it is still useful when examining spherical mechanisms used as components in larger manipulators, as is the case with both the metamorphic parallel manipulator and the metamorphic hand.

In this study, it was found that although more complex in appearance, spherical linkage-based metamorphic mechanisms can have very practical applications and spherical trigonometry is a tool well suited to their study.

References

[1] K. Zhang, J. S. Dai, and Y. Fang, "Geometric constraint and mobility variation of two $3S_v\mathrm{P}S_v$ metamorphic parallel mechanisms," *Journal of Mechanical Design*, vol. 135, no. 1, p. 011001, 2013.

[2] L. Cui and J. S. Dai, "Posture, workspace, and manipulability of the metamorphic multifingered hand with an articulated palm," *Journal of Mechanisms and Robotics, Trans. ASME*, vol. 3, no. 2, p. 021001, 2011.

[3] G. Wei, J. S. Dai, S. Wang, and H. Luo, "Kinematic analysis and prototype of a metamorphic anthropomorphic hand with a reconfigurable palm," *International Journal of Humanoid Robotics*, vol. 08, no. 03, pp. 459–479, 2011.

[4] J. M. McCarthy, *Geometric Design of Linkages*. Springer, 2000.

[5] C. H. Chiang, *Kinematics of Spherical Mechanisms*. Krieger Pub Co, 2000. ISBN: 1-57524-155-2.

[6] J. Duffy, *Analysis of Mechanisms and Robot Manipulators*. 1980.

[7] M. Hirose, "Development of the holonic manipulator and its control," in *Decision and Control, 1990., Proceedings of the 29th IEEE Conference on*, pp. 91–96 vol.1, 1990.

[8] M. Durna, A. M. Erkmen, and I. Erkmen, "The self-reconfiguration of a holonic hand: the holonic regrasp," in *Intelligent Robots and Systems, 2000. (IROS 2000). Proceedings. 2000 IEEE/RSJ International Conference on*, vol. 3, pp. 1993–1998 vol.3, 2000.

[9] G. S. Chirikjian, "Kinematics of a metamorphic robotic system," in *Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*, pp. 449–455 vol.1, 1994.

[10] S. Murata, H. Kurokawa, and S. Kokaji, "Self-assembling machine," in *Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*, pp. 441–448 vol.1, 1994.

[11] S. Murata, H. Kurokawa, E. Yoshida, K. Tomita, and S. Kokaji, "A 3-d self-reconfigurable structure," in *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, vol. 1, pp. 432–439 vol.1, 1998.

[12] M. Wada and H. H. Asada, "A holonomic omnidirectional vehicle with a reconfigurable footprint mechanism and its application to wheelchairs," in *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, vol. 1, pp. 774–780 vol.1, 1998.

[13] G. Hong, W. Zhelong, and W. Hongwei, "Shape reconfigurable mechanism of an earthquake rescue robot," in *Mechatronics and Automation, 2005 IEEE International Conference*, vol. 3, pp. 1145–1150 Vol. 3, 2005.

[14] A. Vertuan, G. Legnani, R. Adamini, D. Tosi, and N. Pedrocchi, "Performance analysis of a reconfigurable redundant parallel manipulator," in *Reconfigurable Mechanisms and Robots, 2009. ReMAR 2009. ASME/IFToMM International Conference on*, pp. 647–655, 2009.

[15] K. Wohlhart, *Kinematotropic Linkages*, ch. 36, pp. 359–368. Springer Netherlands, 1996.

[16] J. S. Dai, "Conceptual study of the dexterous reconfigurable assembly and packaging system," *Science and Technology Report, PS 960326,Unilever Research.*

[17] J. S. Dai and J. R. Jones, "Mobility in metamorphic mechanisms of foldable/erectable kinds," *Journal of Mechanical Design*, vol. 121, no. 3, pp. 375–382, 1999.

[18] J. M. Herve, "The lie group of rigid body displacements, a fundamental tool for mechanism design," *Mechanism and Machine Theory*, vol. 34, no. 5, pp. 719–730, 1999.

[19] J. J. Parise, L. L. Howell, and S. P. Magleby, "Ortho-planar mechanisms," in *26th Biennial Mechanisms and Robotics Conference*, 2000. DETC2000/MECH-14193.

[20] C. Galletti and P. Fanghella, "Single-loop kinematotropic mechanisms," *Mechanism and Machine Theory*, vol. 36, no. 6, pp. 743–761, 2001.

[21] C. Liu and T. Yang, "Essence and characteristics of metamorphic mechanisms and their metamorphic ways," in *11th World Congress in Mechanism and Machine Science*, pp. 1285–1288, 2004.

[22] J. S. Dai and J. R. Jones, "Matrix representation of topological changes in metamorphic mechanisms," *ASME J. Mech. Des.*, vol. 127, no. 4, pp. 837–840, 2005.

[23] J. M. Rico, J. Gallardo, and B. Ravani, "Lie algebra and the mobility of kinematic chains," *Journal of Robotic Systems*, vol. 20, no. 8, pp. 477–499, 2003.

[24] J. M. Rico, L. D. Aguilera, J. Gallardo, R. Rodriguez, H. Orozco, and J. M. Barrera, "A more general mobility criterion for parallel platforms," *Journal of Mechanical Design*, vol. 128, no. 1, pp. 207–219, 2006.

[25] J. S. Dai, Z. Huang, and H. Lipkin, "Mobility of overconstrained parallel mechanisms," *Journal of Mechanical Design*, vol. 128, no. 1, pp. 220–229, 2006.

[26] H.-S. Yan and C.-H. Kuo, "Topological representations and characteristics of variable kinematic joints," *Journal of Mechanical Design*, vol. 128, no. 2, pp. 384–391, 2006.

[27] C.-H. Kuo and H.-S. Yan, "On the mobility and configuration singularity of mechanisms with variable topologies," *Journal of Mechanical Design*, vol. 129, no. 6, pp. 617–624, 2007.

[28] L. Zhang, D. Wang, and J. S. Dai, "Biological modeling and evolution based synthesis of metamorphic mechanisms," *Journal of Mechanical Design*, vol. 130, no. 7, pp. 072303–11, 2008.

[29] L. Zhang and J. S. Dai, "Reconfiguration of spatial metamorphic mechanisms," *Journal of Mechanisms and Robotics*, vol. 1, no. 1, pp. 011012–8, 2009.

[30] Z. Liping and J. S. Dai, "Metamorphic techniques and geometric reconfiguration principles," in *Reconfigurable Mechanisms and Robots, 2009. ReMAR 2009. ASME/IFToMM International Conference on*, pp. 32–40, 2009.

[31] K. Chin-Hsing, J. S. Dai, and Y. Hong-Sen, "Reconfiguration principles and strategies for reconfigurable mechanisms," in *Reconfigurable Mechanisms and Robots, 2009. ReMAR 2009. ASME/IFToMM International Conference on*, pp. 1–7, 2009.

[32] Y. Ting-li, L. An-xin, M. Lu-Zhong, and H. Lu-Bin, "Structure composition principle of reconfigurable mechanisms and basic methods for changing topological structure," in *Reconfigurable Mechanisms and Robots, 2009. ReMAR 2009. ASME/IFToMM International Conference on*, pp. 104–109, 2009.

[33] K. Zhang, Y. Fang, G. Wei, and J. Dai, *Structural Representation of Reconfigurable Linkages*, ch. 13, pp. 127–137. Springer London, 2012.

[34] K. Zhang, E. Emmanouil, Y. Fang, and J. Dai, *Type-Changeable Kinematic Pair Evolved Reconfigurable Parallel Mechanisms*, ch. 28, pp. 309–319. Springer London, 2012.

[35] E. R. Leal and J. S. Dai, "From origami to a new class of centralized 3-DOF parallel mechanisms," *ASME Conference Proceedings*, vol. 2007, no. 48094, pp. 1183–1193, 2007.

[36] D. L. Wang and J. S. Dai, "Theoretical foundation of metamorphic mechanism and its synthesis," *Chinese Journal Mechanical Engineering*, vol. 43, no. 8, pp. 32–42, 2007.

[37] J. S. Dai and D. Wang, "Geometric analysis and synthesis of the metamorphic robotic hand," *Journal of Mechanical Design*, vol. 129, no. 11, pp. 1191–1197, 2007.

[38] C. Lei, J. S. Dai, and W. De Lun, "Workspace analysis of a multifingered metamorphic hand," in *Reconfigurable Mechanisms and Robots, 2009. ReMAR 2009. ASME/IFToMM International Conference on*, pp. 589–595, 2009.

[39] Q. Zhengyan, W. Hongbo, H. Zhen, and Z. Lili, "Kinematics of a quadruped/biped reconfigurable walking robot with parallel leg mechanisms," in *Reconfigurable Mechanisms and Robots, 2009. ReMAR 2009. ASME/IFToMM International Conference on*, pp. 558–564, 2009.

[40] W. Guowu and J. S. Dai, "Geometry and kinematic analysis of an origami-evolved mechanism based on artmimetics," in *Reconfigurable Mechanisms and Robots, 2009. ReMAR 2009. ASME/IFToMM International Conference on*, pp. 450–455, 2009.

[41] K. Zhang and J. Dai, *Kinematics of an Overconstrained 6R Linkage with 2-Fold Rotational Symmetry*, ch. 29, pp. 229–236. Springer Netherlands, 2012.

[42] J. S. Dai and F. Cannella, "Stiffness characteristics of carton folds for packaging," *Journal of Mechanical Design*, vol. 130, no. 2, p. 022305, 2008.

[43] D. W. Carroll, S. P. Magleby, L. L. Howell, R. H. Todd, and C. P. Lusk, "Simplified manufacturing through a metamorphic process for compliant ortho-planar mechanisms," in *ASME International Mechanical Engineering Congress and Exposition*, vol. 118 (1), pp. 389–400, ASME, 2005.

[44] M. Yim, D. Duff, and K. Roufas, "PolyBot: a modular reconfigurable robot," in *Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on*, vol. 1, pp. 514 –520, 2000.

[45] S. Murata and H. Kurokawa, "Self-reconfigurable robots," *Robotics Automation Magazine, IEEE*, vol. 14, pp. 71 –78, march 2007.

[46] E. Yoshida, S. Murata, S. Kokaji, K. Tomita, and H. Kurokawa, "Micro self-reconfigurable robotic system using shape memory alloy," *Distributed autonomous robotic systems*, vol. 4, pp. 145–154, 2000.

[47] J. S. Dai, M. Zoppi, and X. Kong, *Reconfigurable Mechanisms and Robots*, ch. Preface of Reconfigurable Mechanisms and Robots. KC Edizioni, 2009. IEEE Catalog No. CFP0943G-PRT.

[48] J. Dai, M. Zoppi, and X. Kong, *Advances in Reconfigurable Mechanisms and Robots I.* Springer, 2012. ISBN: 978-1-4471-4140-2.

[49] J. S. Dai and R. Jones, "Mobility in metamorphic mechanisms of foldable/erectable kinds," in *Proceedings of the 25th ASME Biennial Mechanisms and Robotics Conference, Atlanta, USA.*, 1998.

[50] G. Gogu, "Branching singularities in kine-matotropic parallel mechanisms," in *Proceedings of the 5th International Workshop on Computational Kinematics, Duisburg, Germany.*, 2009.

[51] K. Zhang, J. S. Dai, Y. Fang, and Q. Zeng, "String matrix based geometrical and topological representation of mechanisms," in *13th World Congress in Mechanism and Machine Science, Guanajuato, Mexico.*, 2011.

[52] R. Claver, "Delta, a fast robot with parallel geometry," in *Proceedings of the International Symposium on Industrial Robot, Switzerland*, pp. 91–100, 1988.

[53] K. H. Hunt, "Structural kinematics of in-parallel-actuated robot-arms," *ASME Journal of Mechanisms, Transmissions and Automation in Design*, vol. 105, pp. 705–712, 1983.

[54] Y. Fang and L. W. Tsai, "Structure synthesis of a class of 4-degreeof freedom and 5-degree of freedom parallel manipulators with identical limb structures," *International Journal of Robotics Research*, vol. 21(9), pp. 799–810, 2002.

[55] D. Zlatanov, I. A. Bonev, and C. M. Gosselin, "Constraint singularities as C-space singularities," in *Advances in Robot Kinematics - Theory and Applications*, pp. 183–192, 2002.

[56] X. Kong, C. M. Gosselin, and P. L. Richard, "Type synthesis of parallel mechanisms with multiple operation modes," *Transactions of ASME: Journal of Mechanical Design*, vol. 129(7), pp. 595–601, 2007.

[57] K. Zhang, J. S. Dai, and Y. Fang, "A new metamorphic mechanism with ability for platform orientation switch and mobility change," in *ASME/ IFToMM International Conference on Reconfigurable Mechanisms and Robots, London, UK*, pp. 626–632, 2009.

[58] K. Zhang, J. S. Dai, and Y. Fang, "Topology and constraint analysis of phase change in the metamorphic chain and its evolved mechanism," *Journal of Mechanical Design*, vol. 132, no. 12, p. 121001, 2010.

[59] D. Gan, J. S. Dai, and Q. Liao, "Mobility change in two types of metamorphic parallel mechanisms," *Journal of Mechanisms and Robotics*, vol. 1, no. 4, p. 041007, 2009.

[60] D. M. Gan, J. S. Dai, and Q. Z. Liao, "Constraint analysis on mobility change of a novel metamorphic parallel mechanism," *Mech. Mach. Theory*, vol. 45(12), pp. 1864–1876, 2010.

[61] C. Melchiorri and M. Kaneko, *Springer Handbook of Robotics*, ch. Robot Hands, pp. 345–360. Springer, 2008.

[62] G. A. Bekey, T. Rajko, and Z. Ilija, "Control architecture for the belgrade/usc hand," *Dextrous robot hands*, pp. 136–149, 1990.

[63] J. Butterfass, M. Grebenstein, H. Liu, and G. Hirzinger, "Dlr-hand ii: next generation of a dextrous robot hand," in *Proceedings of 2001 IEEE International Conference on Robotics and Automation*, vol. 1, pp. 109–114, 2001.

[64] R. Ambrose, H. Aldridge, R. Askew, R. Burridge, W. Bluethmann, M. Diftler, C. Lovchik, D. Magruder, and F. Rehnmark, "Robonaut: Nasa's space humanoid," *Intelligent Systems and their Applications*, vol. 15, no. 4, pp. 57–63, 2000.

[65] C. S. Lovchik and A. D. Myron, "The robonaut hand: A dexterous robot hand for space,," in *Proceedings of 1999 IEEE International Conference on Robotics and Automation*, vol. 2, pp. 907–912, 1999.

[66] R. Walker, "Design of a dextrous hand for advanced clawar applications," in *Conference*

Documentation of the 6th International Conference on Climbing and Walking Robots (CLAWAR 2003), (Catania, Italy), pp. 17–19, 2003.

[67] W. Townsend, "The barrett hand grasper-programmably flexible part handling and assembly," *Industrial Robot: An International Journal*, vol. 27, pp. 181–188, 2000.

[68] G. Carbone and M. Ceccarelli, "Design of larm hand: Problems and solutions," in *Automation, Quality and Testing, Robotics, 2008. AQTR 2008. IEEE International Conference on*, vol. 2, pp. 298–303, 2008.

[69] C. Yao, Z. Qiang, M. Ceccarelli, G. Carbone, Y. Shuangji, and L. Zhen, "Design and simulation of a dsp controller for a larm hand," in *Informatics in Control, Automation and Robotics (CAR), 2010 2nd International Asia Conference on*, vol. 1, pp. 361–364, 2010.

[70] F. Lotti, P. Tiezzi, G. Vassura, L. Biagiotti, G. Palli, and C. Melchiorri, "Development of ub hand 3: Early results," in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pp. 4488–4493.

[71] G. Palli, U. Scarcia, C. Melchiorri, and G. Vassura, "Development of robotic hands: The ub hand evolution," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pp. 5456–5457.

[72] H. Liu, K. Wu, P. Meusel, N. Seitz, G. Hirzinger, M. H. Jin, Y. W. Liu, S. W. Fan, T. Lan, and Z. P. Chen, "Multisensory five-finger dexterous hand: The dlr/hit hand ii," in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pp. 3692–3697, 2005.

[73] C. Dongmin, S. Seunghoon, K. Ja Choon, C. Hyouk Ryeol, and M. Hyungpil, "The skku

hand: Work in progress," in *Ubiquitous Robots and Ambient Intelligence (URAI), 2012 9th International Conference on*, pp. 437–438.

[74] J. S. Dai, "Robotic hand with palm section comprising several parts able to move relative to each other," WO/2005/105391, Nov. 10, 2005 (International Patent No. PCT/GB2005/001665 and UK Patent No. GB04 095 48.5) 2005.

[75] J. S. Dai, D. L. Wang, and L. Cui, "Orientation and workspace analysis of the multifingered metamorphic hand – metahand," *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 942–947, 2009.

[76] G. Wei and J. S. Dai, "Origami-inspired integrated planar-spherical overconstrained mechanisms," *Journal of Mechanical Design, Trans. ASME*, vol. 136, no. 5, p. 051003, 2014.

[77] J. Craig, *Introduction to Robotics: Mechanics and Control*. Addison-Wesley Series in Electrical & Computer Engineering, Addison-Wesley, 1989.

[78] R. M. Murray, Z. Li, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*. 1994.

[79] L. W. Tsai, *Robot analysis: the mechanics of serial and parallel manipulators*. Wiley-Interscience, 1999.

[80] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot modeling and control*. Hoboken, NJ: John Wiley & Sons, 2006. 2005054227 Mark W. Spong, Seth Hutchinson, M. Vidyasagar. ill. ; 25 cm. Includes bibliographical references (p. 459-470) and index.

[81] Y. Liu and K. L. Ting, "On the rotatability of spherical n-bar chains," *Transactions of the ASME: Journal of Mechanical Design*, vol. 116, no. 9, pp. 920–923, 1994.

[82] C. M. Gosselin and J. F. Hamel, "The agile eye: a high-performance three-degree-of-freedom camera-orienting device," in *Proceedings of IEEE International Conference on Robotics and Automation*, (San Diego, CA), pp. 781–787, 1994.

[83] C. W. Wampler, "Displacement analysis of spherical mechanisms having three or fewer loops," *ASME Journal of Mechanical Design*, vol. 126, pp. 93–100, 2004.

[84] K. C. Gupta and A. S. Beloiu, "Branch and circuit defect elimination in spherical four-bar linkages," *Mechanism*, vol. 33, no. 5, pp. 491–504, 1998.

[85] H. Sadjadian and H. D. Taghirad, "Kinematic analysis of the hydraulic shoulder: a 3-DOF redundant parallel manipulator," in *Mechatronics and Automation, 2005 IEEE International Conference*, vol. 3, pp. 1442–1446, 2005.

[86] M. Ceccarelli, *Fundamentals of Mechanics of Robotic Manipulation.* Kluwer/Springer, 2004. ISBN: 1-4020-1810-X.

[87] J. Ziegler and N. Nichols, "Optimum settings for automatic controllers," *trans. ASME*, vol. 64, no. 11, pp. 759–768, 1942.

[88] G. F. Franklin, J. D. Powell, and M. Workman, eds., *Digital Control of Dynamic Systems.* World Student Series, Addison-Wesley, third edition ed., 1998.

[89] K. Ogata, *Modern Control Engineering.* Upper Saddle River, NJ, USA: Prentice Hall PTR, 4th ed., 2001.

[90] E. Emmanouil, K. Zhang, and J. S. Dai, "Control strategy and trajectory planning for reconfiguration of a va based metamorphic parallel manipulator," in *International Design Engineering Technical Conferences*, (International Design Engineering Technical Conferences. 2013 ASME), ASME, 2013.

[91] E. Emmanouil, G. Wei, and J. S. Dai, "Spherical trigonometry constrained kinematics for a dexterous robotic hand with an articulated palm," *Robotica*, vol. FirstView, pp. 1–18, 2015.

[92] G. Wei, V. Aminzadeh, E. Emmanouil, and J. S. Dai, "Structure design, kinematics and grasp constraint of a metamorphic robotic hand for deboning operation," *Proceeding of Design Engineering Technical Conferences & Computers and Information in Engineering Conference, DETC2013-13408*, 2013.